

**Sistemas Informáticos 2009/2010**

**Facultad de Informática**

**Universidad Complutense de Madrid**



# **Geomantes<sup>PC</sup>**

## **Desarrolladores**

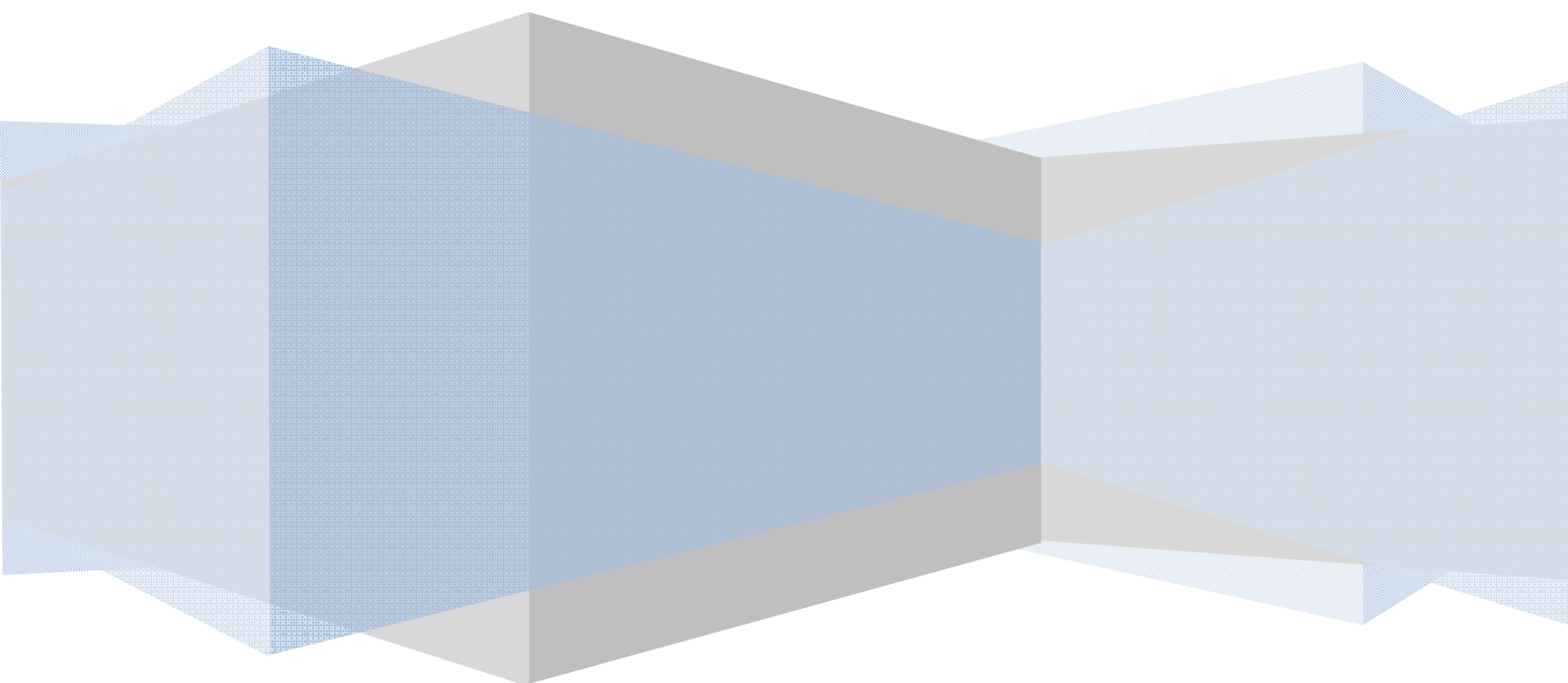
**Alejandro Barrera de Madariaga**

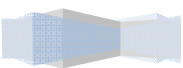
**Alfonso Jiménez Regalado**

**Javier Sangrador Martín**

## **Profesor Director**

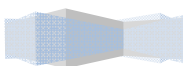
**Fernando Sáenz Pérez**



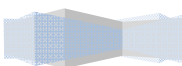


# INDICE

<b>RESUMEN .....</b>	<b>5</b>
<b>ABSTRACT.....</b>	<b>6</b>
<b>DESCRIPCIÓN DEL PROYECTO.....</b>	<b>7</b>
MOTIVACIÓN .....	7
FILOSOFÍA SUBYACENTE.....	7
OBJETIVOS.....	8
HISTORIAL DE VERSIONES .....	9
ESPECIFICACIONES TÉCNICAS.....	12
<b>TECNOLOGÍAS EN USO .....</b>	<b>14</b>
DESCRIPCIÓN .....	14
<i>Chaeron GPS Library</i> .....	14
<i>Google Static Maps API</i> .....	15
<b>EL GPS Y LA GEOLOCALIZACION.....</b>	<b>21</b>
DESCRIPCIÓN .....	21
EL GPS .....	21
LOCALIZACIÓN GEOGRÁFICA Y TIPOS DE COORDENADAS.....	22
<i>Coordenadas Geográficas</i> .....	22
<i>Meridianos</i> .....	22
<i>Paralelos</i> .....	23
<i>Datum</i> .....	24
<i>Datum en Geomantes</i> .....	25
<i>Sistema WGS-84</i> .....	25
<i>Proyecciones</i> .....	26
<i>Proyección UTM (Universal Transverse Mercator)</i> .....	27
GLOSARIO DE TERMINOLOGÍA .....	29
<b>DESARROLLO DEL SISTEMA.....</b>	<b>30</b>
CASOS DE USO .....	30
<i>Navegación</i> .....	31
<i>Rutas</i> .....	32
<i>Mapas</i> .....	33
<i>Pois</i> .....	35
<i>Ayuda</i> .....	37
<i>Calibración</i> .....	38
<i>Configuración</i> .....	39
VIABILIDAD Y ALCANCE .....	41
DISEÑO .....	43
<i>Modelo-Vista-Controlador (MVC)</i> .....	44
<i>Singleton</i> .....	47
<i>Clase Controlador</i> .....	48
<i>Clase Geomantes</i> .....	49
<i>Paquete Modelo</i> .....	50
<i>Paquete Vista</i> .....	50
<i>Paquete Utils</i> .....	50



<b>PRUEBAS.....</b>	<b>51</b>
CALIDAD DEL SOFTWARE .....	51
EJEMPLOS .....	53
<b>MANUAL DE USUARIO.....</b>	<b>54</b>
¿QUÉ ES GEOMANTES <sup>PC</sup> ? .....	54
REQUISITOS HARDWARE .....	54
<i>Dispositivo GPS</i> .....	54
<i>Bluetooth</i> .....	54
<i>Conexión a Internet</i> .....	55
REQUISITOS SOFTWARE .....	55
<i>Sistema Operativo</i> .....	55
<i>ExtGPS 0.31</i> .....	55
<i>Archivos externos</i> .....	55
INSTALACIÓN .....	56
SISTEMA DE MENÚS.....	57
<i>Menú Principal</i> .....	57
<i>Menú Configuración</i> .....	61
<i>Menú Calibración</i> .....	65
<i>Menú Editar Mapas</i> .....	67
<i>Menú Navegación Libre</i> .....	80
FUNCIONAMIENTO DE APLICACIONES EXTERNAS .....	81
<i>ExtGps</i> .....	81
<i>PoiConverter</i> .....	81
<b>CONCLUSIONES .....</b>	<b>83</b>
CONTINUIDAD DEL PROYECTO Y POSIBLES MEJORAS.....	83
EXPERIENCIAS Y CONOCIMIENTOS ADQUIRIDOS .....	85
<b>GLOSARIO.....</b>	<b>86</b>
<b>BIBLIOGRAFÍA.....</b>	<b>87</b>
<b>PALABRAS CLAVE.....</b>	<b>89</b>





## RESUMEN

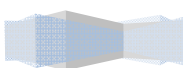
Este documento pretende reflejar el trabajo realizado para la asignatura de Sistemas Informáticos de la Ingeniería Superior de Informática de la Universidad Complutense de Madrid. *GeomantesPC* pasa por ser la translación del navegador GPS *Geomantes*<sup>[1]</sup> basado en mapas gráficos (*raster* o *bitmap*)<sup>[2]</sup> al mundo de los ordenadores personales.

Respetando los conceptos sobre los que se sustenta *Geomantes*<sup>[1]</sup> tanto de **gratuidad** como de máxima **compatibilidad** posible, hemos desarrollado un navegador GPS utilizando en todo momento herramientas *Open source* y que, gracias a la portabilidad de la JVM<sup>[3]</sup> de Sun Microsystems<sup>®</sup>, se posibilita su utilización en cualquier entorno que cuente con esta herramienta. Además, *Geomantes*<sup>PC</sup> se caracteriza por ofrecer una gran **adaptabilidad** a los requisitos del usuario.

A lo largo del desarrollo de la aplicación nos hemos encontrado con una serie de retos que hemos superado satisfactoriamente. Estas dificultades son las que han ido moldeando la forma, y en cierta medida, los contenidos que en esta memoria se explicarán.

- Daremos una explicación detallada de las características de la aplicación, ilustrándolo con diagramas.
- Enumeraremos los casos de uso.
- Hablaremos de las especificaciones y requisitos mínimos.
- Detallaremos a fondo todas las tecnologías involucradas en el desarrollo.

Por último, elaboraremos una conclusión de todo el trabajo realizado y de las metas y objetivos alcanzados, resaltando los aspectos positivos de nuestra aplicación y señalando aquellos en los que se debe continuar trabajando. Expondremos las ideas que consideramos pueden resultar interesantes de cara a futuras mejoras y concluiremos con un pequeño resumen de lo que ha supuesto personal y profesionalmente el desarrollo de *Geomantes*<sup>PC</sup> para los integrantes de este grupo.



## ABSTRACT

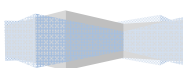
This document pretends to reflect the work done for the subject of Systems Engineering of Computer Science at the Complutense University of Madrid. GeomantesPC goes from Geomantes GPS Navigator<sup>[1]</sup>, based on graphical maps (raster or bitmap)<sup>[2]</sup>, to the world of personal computers.

Concerning topics which Geomantes<sup>[1]</sup> is related, to be **free** for users and to have as much **compatibility** as better, we have developed a GPS navigator using Open Source tools. The portability of the JVM<sup>[3]</sup> Sun Microsystems<sup>®</sup> allows to our program to be used in any environment which provides support for Java application. In addition, GeomantesPC is characterized by offering a high **adaptability** to user requirements.

Throughout the application's deployment we have encountered a number of challenges that we have been overcome successfully. These difficulties are those that have been shaping the project and the contents which are explained in this report.

- We will give a detailed explanation of the features of the application, illustrated with diagrams.
- We will list the use cases.
- We will talk about the specifications and minimum requirements.
- We will detail in depth all the technologies involved in development.

Finally, we will give a conclusion of the whole work performed and the goals and objectives achieved, highlighting the positive aspects of the application and remarking those in which the next developers should continue working. We will discuss the ideas that we believe are interesting for future improvements and we conclude with a brief summary of what has supposed, in the personal and professional development, GeomantesPC for members of this group.



# DESCRIPCIÓN DEL PROYECTO

## MOTIVACIÓN

El desarrollo de sistemas de navegación GPS para automóviles, desde que en 2001 saliera al mercado el primer dispositivo, ha proliferado de manera notable. Impulsados por el avance tecnológico en el mundo de la telefonía móvil, se han desarrollado una serie de navegadores diseñados específicamente para arquitecturas móviles, aprovechando la reciente incorporación de receptores GPS en pequeños terminales.

Sin embargo, dadas las limitaciones técnicas propias de esta clase de dispositivos tales como el reducido tamaño de la pantalla, la limitada memoria interna o los motores de proceso de bajo rendimiento, las características gráficas y de edición de imágenes han sido altamente desatendidas. Además, hemos notado la ausencia de soluciones que aborden de manera profunda y completa el uso *offroad*, pese al gran número de personas que utilizan su tiempo libre para realizar actividades de montaña, como senderismo, mountain bike o travesías.

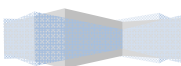
Con el abaratamiento de los ordenadores portátiles y la reciente aparición de los computadoras *Netbook*<sup>[4]</sup> son muchas las posibilidades de aprovechar la combinación de potencia computacional y reducido tamaño.

De aquí nace la idea de crear un navegador GPS para PC, manteniendo las funcionalidades de las versiones para telefonía móvil e incorporando nuevas características que expriman todo el potencial de un ordenador portátil (pantalla amplia, procesador potente, mayores capacidades de almacenamiento, etc.) y que cubra las demandas de todo tipo de usuarios.

*GeomantesPC* ha sido concebido para todos aquellos que busquen una herramienta práctica, fiable y gratuita con la que navegar desde sus ordenadores portátiles.

## FILOSOFÍA SUBYACENTE

Teniendo en cuenta el estado del arte, entendiéndose este como el nivel más alto de desarrollo tecnológico para una materia y un momento dado (en nuestro caso, el software de navegación GPS para ordenadores portátiles existentes hoy en día), hemos tomado como referencia algunas aplicaciones que se sitúan a la vanguardia en cuanto a prestaciones y facilidad de uso, extrayendo las funcionalidades más destacadas e incorporándolas a nuestra aplicación, consiguiendo revalorizar nuestro producto final.







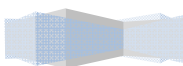
Cabe señalar dos aspectos fundamentales que diferencian a *GeomantesPC* frente a este tipo de aplicaciones: en primer lugar, *GeomantesPC* ha sido concebido como un proyecto *OpenSource* de desarrollo libre, abierto a toda la comunidad y que no pretende competir con este tipo de herramientas, muy orientadas al mundo de los negocios. En la mayoría de los casos, se adquieren bajo costosas licencias limitando su uso a un número reducido de equipos o incluso a uno sólo. Por otro lado, la utilización de *bitmaps* o imágenes *raster* en lugar de mapas vectoriales, nos brindan la oportunidad de ser más flexibles en cuanto a la utilización de mapas, permitiendo tanto el uso de mapas propios como de los descargados desde el servicio de Google Maps, si bien presentan un reto a la hora de trabajar con ellos ya que carecen de información vectorial.

Por todo ello, hemos apostado por crear un producto accesible a todo el mundo, con un cuidado e intuitivo entorno gráfico, donde el usuario satisfaga todas sus necesidades gracias a un amplio conjunto de herramientas y servicios, entre ellos la navegación sobre imágenes digitales previamente calibradas, la edición gráfica de rutas, la exploración de mapas o la edición de puntos de interés propios.

## OBJETIVOS

Enumeramos a continuación una lista con los objetivos prioritarios de nuestra aplicación, que han dirigido el diseño de la aplicación en todo momento:

-  Crear una herramienta accesible a todo el público, gratuita y establecer las estructuras necesarias para que la comunidad de programadores puedan ayudar a mejorarla de forma directa.
-  Reunir características y funcionalidades poco frecuentes en este tipo de aplicaciones, que hagan de esta una herramienta potente e innovadora.
-  Mantener un alto grado de compatibilidad gracias a las posibilidades que ofrece el desarrollo en J2SE.
-  Permitir el uso de la aplicación no sólo como navegador GPS sino también como navegador *offroad*, para aficionados a toda clase de actividades en parajes donde la señal del gps sea limitada.



## HISTORIAL DE VERSIONES

### 6 Octubre 2009

Primera versión de la práctica. Limpia de errores de compilación y éxito al conectar con el simulador de symbian S60 5th Edition SDK.

### 22 Octubre 2009

Investigación sobre compatibilidad de la aplicación con Windows ME. El proyecto GEOMANTES ha sido implementado en J2ME. El sistema operativo para móviles de Microsoft no soporta J2ME, como apunta Javier Cancela en su blog: ["Windows Mobile es la única plataforma móvil importante que no soporta J2ME; "](#). Intento de Conversión del código nativo Java a Visual Basic.NET

### 26 Octubre 2009

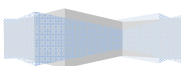
Migración de la aplicación desde J2ME a VB.NET con el asistente Java Language Conversion Assistant que ofrece el Framework de MS, Visual Studio 2003©. Esta es el único medio no de pago para convertir un proyecto desde J2ME a VB.NET. Sin embargo, si existen multitud de aplicaciones para transformar código nativo .NET a J2SE. Desde el principio se observan grandes dificultades.

Tras la conversión se detectan más de 350 problemas de los cuales se consiguen subsanar 22. Los restantes errores están indocumentados en la red y no se encuentra solución. Como muestra se listan las siguientes empresas dedicadas a la creación de proyectos en .NET desde código fuente Java:

[Tometa software,](#)  
[Xen global tech,](#)  
[Xicom technologies](#)

El problema no es sólo reescribir todo el código, cuya tarea no se debe trivializar, si no a continuación investigar el uso del emulador de Windows Mobile y enfrentarse a la cantidad de contratiempos derivados de esto.

Por tiempo y recursos disponibles se considera inviable.



### 20 Noviembre 2009

Investigaciones en el campo de los mapas vectoriales. Intento de lectura y procesado de distintos tipos de mapas vectoriales. Desarrollo de diversas pruebas sobre el procesado de los mapas ofrecidos por <http://www.openstreetmap.org/>

El crecimiento exponencial del número de nodos a procesar en este tipo de representación hace inviable el tratamiento de mapas mínimamente complejos.

Investigación sobre métodos de ingeniería inversa para obtener información legible de otro tipo de mapas vectoriales.

### 1 Diciembre 2009

Primera versión de la interfaz gráfica de la aplicación en su versión para PC

### 11 Diciembre 2009

Primera versión mas o menos estable de la calibración de mapas.

### 27 Diciembre 2009

Obtención de la latitud y longitud exacta del punto medio de un mapa. Esto abre las puertas para el desarrollo de rutas y POIs. Creación de cursores para la depuración del desarrollo que simulan el movimiento del usuario.

### 15 Enero 2010

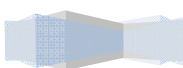
Diversos arreglos en la clase controlador. Funcionamiento de la versión simulada mediante cursores de la Navegación libre en modo GoogleMaps.

### 25 Febrero 2010

Cambio de interfaz de la aplicación a una más vistosa con iconos y sistema de pantallas emergentes mediante la pulsación de botones. Añadida funcionalidad para poder cargar mapas desde memoria.

### 6 Marzo 2010

Añadido el panel de grabación de rutas. Añadida funcionalidad que permite hacer zoom sobre el mapa. Separación en 2 menús distintos de la previsualización del mapa y su navegación.



### 15 Marzo 2010

Depurado el sistema de calibración para hacerlo más intuitivo. Permitida la rectificación de los puntos antes de calibrar. Añadida la edición grafica de rutas. Añadida la traducción al ingles de diversos elementos de la aplicación.

### 15 Abril 2010

Grabación rutas con el movimiento del usuario. Permitido deshacer al pintar rutas. Eliminación del error acumulado durante el movimiento. Nueva versión de la interfaz gráfica. Funcionalidad añadida guardar mapa.

### 2 Mayo 2010

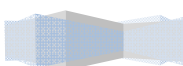
Añadidas distintas vistas en el modo google maps. Añadida funcionalidad guardar rutas. Funcionando la navegación GPS con dispositivo real. Creación de la pestaña POI

### 19 Mayo 2010

Versión estable de la funcionalidad añadir POI. Añadida funcionalidad *Ver lugar*

### 20 Mayo 2010 en adelante

Etapas intensivas de depuración de los casos de uso y errores surgidos durante la marcha. Incluidas mejoras visuales y mejoras en la usabilidad.



## ESPECIFICACIONES TÉCNICAS

Las herramientas utilizadas para el desarrollo han sido:

- OziExplorer [versión 3.95.4] (<http://www.ozexplorer.com>)

Es un sistema de navegación 'offroad' caracterizado porque no es vectorial al contrario que otros programas como TomTom. OziExplorer no permite indicar la posición del vehículo dentro de las vías al no ser un programa de cartografía vectorial. TomTom asignará nuestra posición al camino más cercano mediante una especie de "imantación". OziExplorer, en cambio, al hacer uso de cartografía en imagen o RASTER, navegará a través de imágenes cartográficas (u ortomapas) indicándonos la posición exacta que marca la señal GPS.

- Adobe Photoshop CS4 (<http://www.adobe.com/es/products/photoshop>).

Proporciona las utilidades necesarias para realizar los retoques fotográficos pertinentes.

- POIConverter [versión 3.95.4]  
(<http://rjdavies.users.btopenworld.com/html/poiconverter.html>)

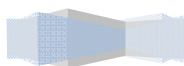
Aplicación gratuita que permite pasar desde los principales formatos de puntos de interés, al estándar CSV que utilizamos en nuestra aplicación. Como herramienta complementaria ha sido de utilidad a la hora de comprobar el correcto funcionamiento de nuestro sistema de POIs.

- NetBeans IDE [version 6.7.1](<http://netbeans.org/>)

Entorno de desarrollo integrado (IDE) para crear aplicaciones de cualquier tipo. La experiencia positiva acumulamos todos los integrantes de este proyecto, junto con las facilidades que se plantean a la hora de programar y la multitud de funcionalidades gráficas presentes, han hecho que nos decantásemos por este entorno.

- CollabNet Subversion Server [version 1.6.9]  
(<http://www.open.collab.net/products/subversion/>)

Utilidad para mantener comunicación entre los desarrolladores y mantener actualizada la aplicación.





- Google Code (<http://code.google.com/>).

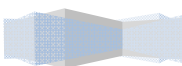
El repositorio para código fuente de Google nos ha servido como lugar de almacenaje de todas y cada una de las versiones de nuestra aplicación.

- Microsoft Word [version 2007]

Herramienta de procesamiento de textos.

- Skype<sup>TM</sup> [version 4.1]( <http://www.skype.com/intl/es/home/>)

Programa gratuito para la comunicación entre los integrantes del grupo y pieza clave del buen entendimiento y desarrollo conjunto.



# TECNOLOGÍAS EN USO

## DESCRIPCIÓN

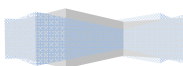
Para nuestro grupo, uno de los principales escollos que hemos tenido que superar ha sido la conectividad de nuestra aplicación con el dispositivo GPS. Al no tratarse de una aplicación para dispositivos móviles, por ejemplo, no es posible el empleo del API de localización JSR 179, al estar diseñado para J2ME.

El número de librerías para J2SE que implementan servicios de localización es ciertamente escaso, sin embargo, hemos encontrado un paquete de distribución libre que satisface nuestras necesidades, cumple con la filosofía de Geomantes<sup>PC</sup> de máxima compatibilidad y que describimos a lo largo de este capítulo.

### Chaeron GPS Library

Provee una interfaz Java de alto nivel para dispositivos GPS que encapsula todas las tareas y procedimientos relativos a la comunicación entre nuestra aplicación y los terminales GPS. Permite el uso de un completo conjunto de funciones sin necesidad para el programador de conocer los detalles técnicos de las mismas. Para entender su funcionamiento, resumimos brevemente el conjunto de funciones más relevantes de esta librería y a continuación, documentamos los métodos mediante los que hemos implementado su uso:

- ◆ `public static GPS newInstance( int protocol ) throws GPSProtocolException`: método estático que devuelve una instancia concreta de GPS. Lanza Excepción cuando el protocolo especificado no se reconoce como válido.
- ◆ `public void addGPSListener( GPSListener listener, int types )`: añade a la estructura GPS un listener asociado a una lista de eventos (definidos en el parámetro types).
- ◆ `public final void connect() throws GPSConnectionException, GPSSerialPortException`: Establece connexion con el GPS a través de un puerto serie. Lanza Excepción si hay algún problema con el dispositivo GPS o el puerto serie por el que se intenta la conexión no está disponible.



- ◆ `public void poll() throws GPSSerialPortException`: recoge los paquetes de datos enviados desde el aparato GPS a través de un puerto serie. Si este último no está disponible, lanza Excepción.
- ◆ `public final void disconnect() throws GPSSerialPortException, GPSSerialPortException`: Desconexión y fin de la comunicación.

### MÉTODOS PROPIOS DE APLICACIÓN DE ESTOS SERVICIOS

`public GPS initGPS()`: Este es el método empleado para iniciar la conexión y recibir la información del dispositivo GPS. En él se detallan aspectos fundamentales para la comunicación como el protocolo de transmisión, que en nuestro caso utilizamos el protocolo NMEA<sup>[6]</sup>, o la lista de eventos que se desean recibir. Además, se establecen el puerto serie para la comunicación, la tasa de ruido permitida o el permiso para la recepción de paquetes de datos incompletos (caso de no tener el bloqueo del satélite), entre otros parámetros.

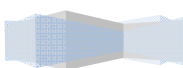
`public void pollGPS()`: Encuesta a la unidad GPS por los datos obtenidos y procesa todos los eventos que se encuentren a la espera de esos datos. La lista de eventos se define en el método `initGPS()`.

`public boolean onGPSEvent(GPSEvent event)`: Listener encargado de capturar y dar respuesta a los eventos lanzados por el método `poll()`. Principalmente nuestra aplicación maneja eventos de cambio de posición, generando una serie de actualizaciones de los datos almacenados y de la representación gráfica de los mismos. El valor devuelto se interpretará como verdadero, si el evento capturado ha sido manejado correctamente o falso si no ha podido serlo y se ha propagado al listener subsiguiente.

`public void disconnectGPS()`: Utilizamos este método para detener el tráfico de datos entre el aparato GPS y nuestra aplicación. La instancia GPS que hubiéremos adquirido previamente mediante `initGPS()` deja de estar disponible.

## Google Static Maps API

Una de las tecnologías existentes que nos ha resultado de mayor utilidad durante el desarrollo ha sido el servicio para programadores que ofrece Google. Este servicio permite descargar, a través de una solicitud HTTP estándar, imágenes estáticas de cualquier zona del mundo a partir de los parámetros URL indicados desde la aplicación. Además, alterando algunos de los parámetros de la llamada se pueden conseguir cambios significativos en la imagen obtenida. A continuación se detallan las



características del mapa que Google permite modificar y hemos decidido incluir en la aplicación:

- ✓ Especificación de ubicaciones: elección de la zona a visualizar mediante coordenadas geográficas o nombres de lugares.
- ✓ Nivel de acercamiento: 4 niveles de acercamiento posibles.
- ✓ Distintas vistas del mapa: mapa de carreteras, satélite, híbrido o terreno.

Para la creación de rutas e inserción de puntos de interés hemos preferido desarrollar nuestro propio código al no adaptarse -el servicio ofrecido por Google- a nuestras necesidades.

Entrando en detalles técnicos, los métodos que codifican la obtención del mapa son los siguientes:

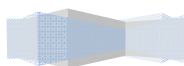
```
public BufferedImage retrieveStaticImage(int width, int height, double lat,
    double lng, int zoom, String format, String mapType) throws Exception {

    return ImageIO.read(new URL(getMapUrl(width, height, lng, lat, zoom, format, mapType)));
}
```

```
String getMapUrl(int width, int height, double lng, double lat, int zoom, String format, String mapType) {
    return "http://maps.google.com/staticmap?center=" +
        lat + "," + lng + "&format=" + format + "&zoom=" + zoom + "&size=" +
        width + "x" + height + "&sh=15" + "&maptype=" + mapType + "&key=" + apiKey;
}
```

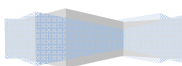
Examinando más a fondo el código se puede apreciar que en la función **getMapUrl** es donde se lleva a cabo directamente la solicitud de la imagen a la página de Google Maps. Los parámetros de entrada de la función definen las características de la imagen a descargar. Como se puede prácticamente adivinar estos parámetros corresponden a:

- ✚ Width y Height: alto y ancho de la imagen
- ✚ Lng y Lat: longitud y latitud (coordenadas geográficas)
- ✚ Zoom : nivel de acercamiento de la imagen
- ✚ Format: formato del archivo ( en nuestro caso .PNG)
- ✚ mapType: distintas vistas ( mapa de carreteras, satélite, híbrido o terreno)



En cuanto a la forma de almacenar en memoria las imágenes obtenidas, hemos decidido mantener la estructura matricial utilizada por los compañeros que realizaron la primera versión del proyecto por resultarnos una manera muy acertada e ingeniosa de resolver el problema de la obtención de nuevos mapas a medida que se aprecia movimiento en el sistema. En el proceso de adaptación de la aplicación a una nueva coyuntura tecnológica hemos decidido aumentar el tamaño de las imágenes de cada celda al disponer de mayor ancho de banda de la conexión, tamaño de memoria y pantalla para la visualización.

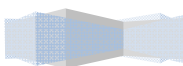
Para una mejor comprensión del proceso anteriormente descrito dejamos de la propia mano de nuestros compañeros la explicación del funcionamiento del sistema por ellos ideado por no querer caer en el error de querer repetir sin necesidad la explicación que ellos ya redactaron:



Como situación inicial, nos encontramos con nueve mapas cargados y perfectamente centrados en el punto en el que nos encontramos.



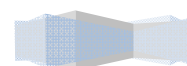
Al realizar un desplazamiento lateral, nos movemos hacia la derecha (hacia la izquierda es simétrico, por lo que no se explica) y los tres fragmentos de mapa correspondientes a los números 1, 4 y 7 dejan de sernos útiles.





Finalmente, eliminamos los mapas 1, 4 y 7 de nuestra estructura y cargamos tres mapas por el lado derecho (simétricamente, izquierdo), de tal forma que tenemos todos los laterales ya cargados por si el usuario se desplaza fuera del mapa central.







# EL GPS Y LA GEOLOCALIZACION

## DESCRIPCIÓN

El principal reto a la hora de comenzar a planificar Geomantes era conseguir conectar de una forma eficiente y simple nuestra aplicación a un dispositivo GPS. Pese a que esa tarea no implica grandes complicaciones, descubrimos que debíamos adquirir un gran número de conceptos que están asociados al mundo de la geolocalización y la navegación con GPS. Todos estos conocimientos, así como su aplicación en nuestro proyecto, es lo que queda reflejado en las siguientes páginas.

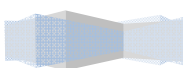
## EL GPS

Las siglas de GPS responden a Sistema de Posicionamiento Global (o Global Positioning System). Gracias a este, podemos determinar la posición de cualquier tipo de objeto con una gran precisión. El sistema GPS está compuesto por tres componentes fundamentales.

- Red de satélites: Cubren toda la superficie de la tierra repartidos en 6 planos con 4 satélites en cada uno (para un total de 24 satélites). Se alimentan gracias a los paneles solares que los constituyen. Tienen una vida útil media de 7 años y medio y se encuentran a una altitud de 20.200 km sobre el nivel del mar.
- Estaciones terrestres: Se intercomunican con los satélites para mantener operativa la estructura.
- Terminales receptores: Más conocidas como Unidades GPS, permiten conocer la posición en la que se encuentran.

La posición del terminal se mide calculando la distancia entre este y los satélites. Esto se hace midiendo el tiempo de respuesta hasta que alcanza el receptor.

La fiabilidad de la señal obtenida por un receptor varía en torno a una serie de parámetros; las nubes, montañas o edificios presentan un problema ineludible que puede reducir la precisión de la señal considerablemente. En condiciones favorables (7, 8 o 9 satélites visibles), se obtienen altísimas precisiones de entorno a 2 metros con una probabilidad superior al 95%. En la siguiente tabla se observan las fuentes de error más comunes y su efecto asociado:



Fuente	Efecto
Ionosfera	$\pm 5$ metros
Efemérides	$\pm 2.5$ metros
Reloj Satelital	$\pm 2$ metros
Distorsión multibanda	$\pm 1$ metro
Troposfera	$\pm 0.5$ metros
Errores numéricos	$\pm 1$ metro (o inferior)

## LOCALIZACIÓN GEOGRÁFICA Y TIPOS DE COORDENADAS

La localización geográfica de un punto se puede realizar detallando uno de estos dos parámetros:

- Coordenadas geográficas en formato Longitud-Latitud.
- Coordenadas UTM (Universal Transverse Mercator).

Asimismo, para poder representar un punto en un mapa con sus valores de coordenadas, es necesario emplear un modelo matemático conocido como *Datum*.

### Coordenadas Geográficas

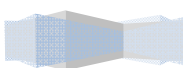
Las coordenadas geográficas son una forma de designar un punto sobre la superficie terrestre teniendo en cuenta dos medidas: longitud y latitud. El formato es el siguiente:

**3°45'25" W**  
**32°11'44" N**

Estas medidas representan la distancia en grados, minutos y segundos con respecto al meridiano de Greenwich en el caso de la longitud, y con respecto al ecuador en el caso de la latitud.

### Meridianos

Se definen los meridianos como las líneas de intersección con la superficie terrestre de los infinitos planos que contienen el eje de la Tierra.

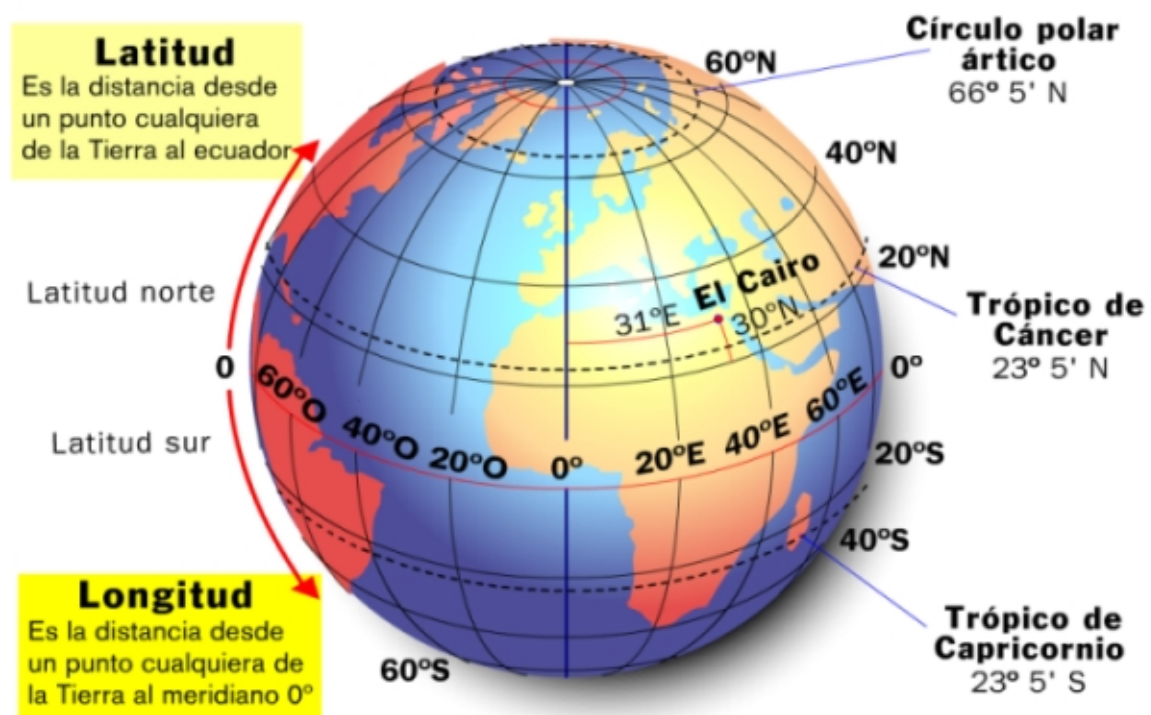


La existencia del meridiano 0º o meridiano de Greenwich divide al globo terráqueo en dos zonas: las situadas al oeste (W) del meridiano 0º hasta el antimeridiano, y las situadas al este (E) del meridiano 0º hasta el antimeridiano.

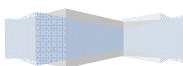
## Paralelos

Se definen los paralelos como las líneas de intersección de los infinitos planos perpendiculares al eje terrestre con la superficie de la Tierra.

Al paralelo de mayor radio se le denomina ecuador. Este paralelo principal divide a la Tierra en dos hemisferios: el norte y el sur. Paralelos geoméricamente a él, se trazan el resto de paralelos, de menor radio, tanto en dirección al polo norte como al polo sur.



*Representación gráfica de la división de la Tierra en meridianos y paralelos*



## Datum

La Tierra no es esférica, ni siquiera es un cuerpo regular achatado por los polos. Esta irregularidad hace que cada país, o incluso cada región, escoja el modelo de cuerpo (definible matemáticamente) que más se ajuste a la forma de la tierra en su territorio. Este cuerpo suele ser un elipsoide.

Los diferentes elipsoides se diferencian unos de otros en sus parámetros, entre los que se encuentran:

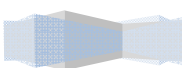
- El radio mayor y menor del elipsoide. (a y b).
- El aplastamiento del elipsoide ( $1/f = 1 - (b/a)$  ).

Cada datum está compuesto por:

- Un elipsoide.
- Un punto llamado "Fundamental" en el que el elipsoide y la tierra son tangentes. De este punto se han de especificar longitud, latitud y el acimut de una dirección desde él establecida.

En el punto Fundamental, las verticales de elipsoide y tierra coinciden. También coinciden las coordenadas astronómicas (las del elipsoide) y las geodésicas (las de la tierra).

Definido el Datum, ya se puede elaborar la cartografía de cada lugar, pues se tienen unos parámetros de referencia.



## Datum en Geomantes

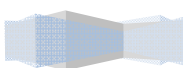
Nuestra aplicación es capaz de interpretar mapas calibrados en distintos Datum, en concreto en los que se presentan a continuación en un fragmento de código fuente:

```
public static Ellipsoid ellipsoid[] = { // id, nombre del
    elipsoide, radio ecuatorial, cuadrado de la excentricidad
    new Ellipsoid(0, "Placeholder", 0.0, 0.0),
    new Ellipsoid(1, "Airy", 6377563, 0.00667054),
    new Ellipsoid(2, "Australian National", 6378160,
0.006694542),
    new Ellipsoid(3, "Bessel 1841", 6377397, 0.006674372),
    new Ellipsoid(4, "Bessel 1841 (Nambia)", 6377484,
0.006674372),
    new Ellipsoid(5, "Clarke 1866", 6378206, 0.006768658),
    new Ellipsoid(6, "Clarke 1880", 6378249, 0.006803511),
    new Ellipsoid(7, "Everest", 6377276, 0.006637847),
    new Ellipsoid(8, "Fischer 1960 (Mercury)",
6378166, 0.006693422),
    new Ellipsoid(9, "Fischer 1968", 6378150, 0.006693422),
    new Ellipsoid(10, "GRS 1967", 6378160, 0.006694605),
    new Ellipsoid(11, "GRS 1980", 6378137, 0.00669438),
    new Ellipsoid(12, "Helmert 1906", 6378200, 0.006693422),
    new Ellipsoid(13, "Hough", 6378270, 0.00672267),
    new Ellipsoid(14, "International", 6378388, 0.00672267),
    new Ellipsoid(15, "Krassovsky", 6378245, 0.006693422),
    new Ellipsoid(16, "Modified Airy", 6377340, 0.00667054),
    new Ellipsoid(17, "Modified Everest", 6377304,
0.006637847),
    new Ellipsoid(18, "Modified Fischer 1960", 6378155,
0.006693422),
    new Ellipsoid(19, "South American 1969", 6378160,
0.006694542),
    new Ellipsoid(20, "WGS 60", 6378165, 0.006693422),
    new Ellipsoid(21, "WGS 66", 6378145, 0.006694542),
    new Ellipsoid(22, "WGS 72", 6378135, 0.006694318),
    new Ellipsoid(23, "WGS 84", 6378137, 0.00669438),
    new Ellipsoid(24, "European 1950", 6378388, 0.00672267)
};
```

Como se puede observar, Geomantes proporciona una alta flexibilidad a la hora de trabajar con distintos modelos matemáticos de georreferenciación.

## Sistema WGS-84

Es el sistema de referencia terrestre adoptado por el Departamento de Defensa de los Estados Unidos para el posicionamiento GPS. WGS-84 es un sistema de coordenadas geocéntrico global basado originariamente en observaciones Doppler del sistema de satélites TRANSIT.



El elipsoide de referencia de WGS-84 es esencialmente el del Sistema Geodésico de referencia 1980 (GRS 80), de la Unión Geodésica y Geofísica Internacional, con cambios menores, sólo en su aplastamiento. Los parámetros del elipsoide WGS-84 son:

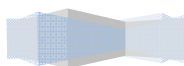
Semieje mayor  $a = 6.378.137$  m. y Aplastamiento  $f = 1/298,257223563$

A día de hoy, la gran mayoría de los receptores GPS generan internamente los datos en el sistema WGS-84.

## Proyecciones

La proyección cartográfica permite representar una superficie esférica como la Tierra en una lámina de papel plana. Una proyección cartográfica es una representación sistemática de los paralelos y meridianos de una superficie tridimensional en una superficie bidimensional. Dado que una superficie plana no puede ajustarse a una esfera sin estirarse o encogerse tampoco es posible representar atributos de un globo en un mapa sin causar distorsiones.

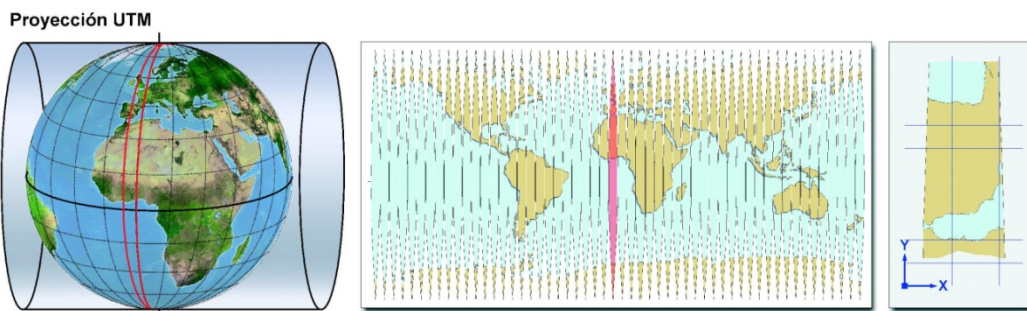
Existen diversas proyecciones y cada una de ellas trata de minimizar las distorsiones. Las proyecciones que se utilizan en la actualidad se han derivado a partir de modelos matemáticos del globo terrestre y todas ellas comparten la misma característica: mostrar la posición correcta de las líneas de longitud y latitud del Planeta. En otras palabras, cada proyección es solamente un reordenamiento de los meridianos y paralelos trasladados del Globo Terrestre a un mapa. Dado que no hay forma de eliminar los errores al trasladar una superficie curva (Tierra) a una superficie plana (mapa) ninguna proyección es geométricamente perfecta. En síntesis, cada proyección es elaborada a partir de una figura geométrica con un propósito particular y por ende tiene sus propias virtudes y limitaciones.



## Proyección UTM (Universal Transverse Mercator)

Para calcular distancias de un punto geográfico a otro dadas sus longitudes y latitudes, debemos llevar a cabo una conversión a coordenadas UTM. La proyección UTM (Universal Transverse Mercator) tiene su origen en el desarrollo cilíndrico conforme de Gauss.

La superficie de referencia utilizada es un elipsoide de revolución al cual se le situará tangente exteriormente un cilindro transverso. La proyección se divide en 60 husos de 6º de longitud que se empiezan a numerar desde el antimeridiano origen, de forma que el meridiano de Greenwich define el límite entre los husos 30 (al oeste) y 31 (al este). España está, de este modo, comprendida entre los husos 28 al 31. El origen de coordenadas Y es el ecuador mientras que las coordenadas X se calculan desde el meridiano central, al que se le asigna un valor de 500000 metros con el fin de evitar coordenadas negativas.

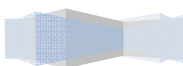


*Diagrama explicativo de la proyección UTM*

## OZIEXPLORER Y LOS FICHEROS .MAP

La geolocalización de los mapas cartográficos solo es posible si asociamos a las imágenes un fichero con la información necesaria. Nuestra elección por motivos de compatibilidad ha sido la de usar el estándar de OziExplorer.

OziExplorer es un programa dedicado al mundo de GPS. Tiene muchas ventajas. Una de ellas es que es el único programa que hoy día te permite trabajar con los GPS de los modelos Garmin, Lowrance, y Magellan, permitiendo el intercambio de datos (rutas, tracks y waypoints) entre la práctica totalidad de los usuarios de GPS. Otros aspectos a destacar es su versatilidad (usa mapas escaneados o digitales), importa mapas, crea rejillas (grids), tiene implementada la función de mapa móvil y la de auto-pilotaje, y un largo etcétera. Además, es extraordinariamente preciso, y sobre todo, posee el mejor apoyo técnico de todos los programas. Su autor, Des Newman, contestará a todas tus preguntas, e incluirá tus sugerencias.



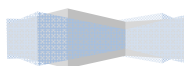
Dicho programa utiliza la extensión “.map” para identificar a los archivos de geolocalización. A continuación se expone un ejemplo y se detallan algunas de sus características:

```
OziExplorer Map Data File Version 2.2
Casa.jpg
Casa.jpg
1 ,Map Code,
WGS 84,, 0.0000, 0.0000,WGS 84
Reserved 1
Reserved 2
Magnetic Variation,,,E
Map Projection,(UTM) Universal Transverse
Mercator,PolyCal,No,AutoCalOnly,No,BSBUseWPX,No
Point01,xy,0,0,in, deg, , ,N, , ,W, grid, 30, 442259.0041454903,
4479280.670948872,N
Point02,xy,2048,0,in, deg, , ,N, , ,W, grid, 30, 444307.0006342536,
4479280.654703642,N
Point03,xy,2048,2048,in, deg, , ,N, , ,W, grid, 30,
444306.98426401743, 4477232.658259679,N
Point04,xy,0,2048,in, deg, , ,N, , ,W, grid, 30, 442258.9877889765,
4477232.674504637,N
Projection Setup,,,,,,,,
Map Feature = MF ; Map Comment = MC These follow if they exist
Track File = TF These follow if they exist
Moving Map Parameters = MM? These follow if they exist
MM0,Yes
MMPNUM,4
MMPXY,1,0,0
MMPXY,2,2048,0
MMPXY,3,2048,2048
MMPXY,4,0,2048
MMPLL,1,-3.6810547207149833,40.46219122467416
MMPLL,2,-3.6569001464900044,40.46233087535915
MMPLL,3,-3.656720698347238,40.44388149386048
MMPLL,4,-3.6808686681269895,40.443741933683825
MM1B,1.0
MOP,Map Open Position,0,0
IWH,Map Image Width/Height,2048,2048
```

→ Sistema de coordenadas utilizado (en este caso, WGS 84).

} Información de las cuatro esquinas que delimitan el mapa.

} Coordenadas asociadas a las esquinas del mapa





## GLOSARIO DE TERMINOLOGÍA

**Datum:** Dadas las irregularidades de la superficie de la tierra, cada país ha de escoger un modelo que se ajusta a la forma de su territorio. Normalmente se suele tratar de un elipsoide. Cada datum se compone por este elipsoide y un punto denominado “Fundamental”, en el que la tierra y el elipsoide son tangentes. De esta forma, definido el datum, tenemos los datos necesarios para elaborar la cartografía de un lugar.

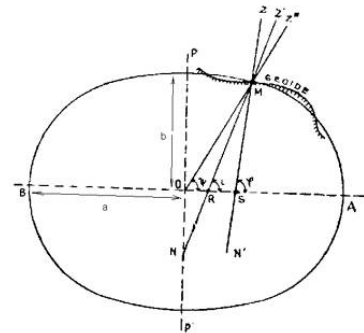


Fig. 8

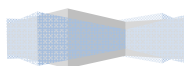
**WGS84:** Son las siglas de World Geodetic System 1984. Sistema de coordenadas mundiales, que data de 1984, que es la base para sistemas de posicionamiento globales como el GPS. Está pendiente una próxima revisión para 2010.

El WGS84 usa como elipsoide de referencia a WGS 84, que es perfectamente intercambiable con el sistema de referencia europeo ETRS 89, estribando su única diferencia en varias décimas de milímetro para el caso de los semiejes menores. [18]

**ED50:** Para el oeste europeo encontramos un tipo de datum denominado ED50 (European Datum 1950) de después de la II Guerra Mundial. Surge por la necesidad de unificar los datos de mapas de países como Alemania, Holanda, Bélgica o Francia, que tenían posicionamientos de latitud y longitud incompatibles. Este sistema tiene su centro al sur de Alemania, donde se situó el centro de la Europa occidental a finales de la guerra fría.

**Raster:** cartografía en imagen o RASTER es la navegación a través de imágenes cartográficas (u ortomapas) indicándonos la posición exacta que marca la señal GPS al contrario de sistemas como TomTom, que asignan nuestra posición al camino más cercano mediante una especie de “imantación”.

**Huso:** Geográficamente los husos horarios son cada una de las veinticuatro áreas en que se divide la Tierra. Se llaman así porque tienen forma de huso de hilar o de gajo de naranja y están centrados en meridianos de una longitud que es múltiplo de 15°.



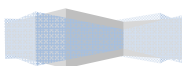
# DESARROLLO DEL SISTEMA

## CASOS DE USO

Un modelo de caso de uso es una especificación de funcionalidad describiendo la interacción del sistema con los actores a veces apoyada en una representación gráfica para facilitar la comprensión pero construida principalmente a partir de texto. Consta normalmente de las siguientes partes:

- Código: Abreviatura que simplifica las referencias posteriores al CU.
- Nombre: Identificador comprensible.
- Actores: Genéricamente se trata de un representante de un papel en un guión. En nuestro caso usamos esa metáfora para describir a un agente externo que interactúa con el sistema. Pese a lo anterior, ciertas cosas generalmente vistas como externas al sistema no son considerados buenos actores. Los sistemas de persistencia como las bases de datos o los archivos, si bien externos, no son actores.
- Descripción: Breve explicación acerca de lo que ocurre durante el CU.
- Precondición: Eventos que han de cumplirse antes del CU.
- Postcondición: Acciones que se cumplen una vez finalizado el CU.
- Flujo principal: Orden de acciones que se completan durante la CU en una ejecución sin imprevistos. Solo se incluye en caso de contener información relevante.
- Flujo alternativo: Posibles desvíos sobre el flujo principal. Solo se incluye en caso de contener información relevante.

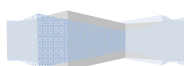
Aclaración: hemos supuesto que el lenguaje seleccionado es el español. Las palabras en *cursiva* dependerán del idioma seleccionado.



## Navegación

<b>CU-0101</b>	<b>Navegación libre en Google Maps</b>
Actores	Usuario.
Descripción	Navegación mediante GPS con mapas del servicio de Google Maps.
Precondición	Disponer de un dispositivo GPS y conexión a internet.
Postcondición	Se permite la libre navegación.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar el idioma.</li> <li>2. Seleccionar Navegación libre.</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>1.1. Si el dispositivo GPS no funciona, se aborta la secuencia.</li> <li>1.2. Si no se dispone de conexión a internet, se aborta la secuencia.</li> </ol>

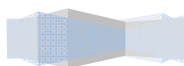
<b>CU-0102</b>	<b>Navegación libre sobre mapas cargados desde archivos</b>
Actores	Usuario.
Descripción	Navegación mediante GPS con mapas almacenados en archivos.
Precondición	Disponer de algún mapa previamente almacenado, además de acceso a un dispositivo GPS.
Postcondición	Se permite la libre navegación.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar el idioma.</li> <li>2. Seleccionar Configuración.</li> <li>3. En el cuadro de opciones, marcar Archivos y <i>Aceptar</i>.</li> <li>4. Seleccionar Navegación libre.</li> <li>5. Seleccionar <i>Cargar mapa</i>.</li> <li>6. Navegar por el sistema de archivos hasta encontrar el mapa que deseamos cargar.</li> <li>7. Seleccionar un mapa para cargar.</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>5.1 Si el dispositivo GPS no funciona, se aborta la secuencia.</li> <li>5.1 Si decidimos <i>salir</i> volvemos a la pantalla principal</li> <li>7.1. Si el mapa cargado no tiene un formato reconocido, volvemos al paso 5.</li> </ol>



## Rutas

<b>CU-0103</b>	<b>Dibujar una ruta y almacenarla</b>
Actores	Usuario.
Descripción	Crear una ruta de manera gráfica, marcando los puntos sobre un mapa.
Precondición	Disponer de algún mapa.
Postcondición	Creación de una ruta.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar el idioma.</li> <li>2. Seleccionar <i>Editar mapas</i>.</li> <li>3. Seleccionar pestaña <i>Rutas</i>.</li> <li>4. Seleccionar <i>Dibujar ruta</i>.</li> <li>5. Marcar en el mapa los puntos de la ruta.</li> <li>6. Seleccionar <i>Guardar ruta</i>.</li> <li>7. Explorar en el sistema de archivos hasta localizar la ubicación deseada.</li> <li>8. Asignar un nombre y guardar la ruta.</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>5.1 Si se marca un punto erróneamente, seleccionar <i>Deshacer último cambio</i>.</li> <li>5.2 Para abortar seleccionar <i>Borrar ruta</i>.</li> </ol>

<b>CU-0104</b>	<b>Grabar una ruta al navegar en GM</b>
Actores	Usuario.
Descripción	Creación de una ruta que memoriza el recorrido mientras se navega.
Precondición	Disponer de un dispositivo GPS.
Postcondición	Ruta con el recorrido realizado.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar Navegación libre.</li> <li>3. Seleccionar pestaña <i>Rutas</i>.</li> <li>4. Seleccionar <i>Grabar ruta</i> para comenzar la grabación</li> <li>5. Seleccionar <i>Guardar ruta</i> para finalizar el recorrido grabado.</li> <li>6. Navegar por el sistema de archivos y seleccionar el directorio de destino.</li> <li>7. Asignar un nombre y guardar la ruta</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>4.1 Para abortar la grabación, seleccionar <i>Borrar ruta</i>.</li> </ol>

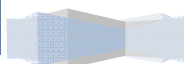


<b>CU-0105</b>	<b>Cargar ruta</b>
Actores	Usuario.
Descripción	Representación gráfica de una ruta almacenada sobre un mapa propio
Precondición	Disponer de una ruta
Postcondición	Se muestra la ruta sobre el mapa.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar Navegación libre.</li> <li>3. Seleccionar pestaña Rutas.</li> <li>4. Seleccionar Cargar ruta.</li> <li>5. Navegar por el sistema de archivos hasta encontrar la ruta especificada.</li> <li>6. Seleccionar la ruta para cargar.</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>4.1 Si el archivo seleccionado no se trata de una ruta, se vuelve al paso 4.</li> </ol>

<b>CU-0106</b>	<b>Cambiar los colores de la ruta</b>
Actores	Usuario.
Descripción	Modificar los colores de una ruta
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar el idioma.</li> <li>2. Seleccionar <i>Editar mapas</i>.</li> <li>3. Seleccionar pestaña Rutas.</li> <li>4. Elegir los tres colores de la ruta</li> </ol>

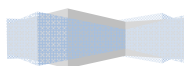
## Mapas

<b>CU-0106</b>	<b>Cambiar vista en mapas de GM</b>
Actores	Usuario.
Descripción	Elección entre las cuatro distintas vistas que ofrece Google Maps.
Precondición	Conexión a Internet disponible.
Postcondición	Vista del mapa en el formato seleccionado
Flujo principal	<ol style="list-style-type: none"> <li>1. Elegir idioma.</li> <li>2. Seleccionar <i>Navegación libre</i> o <i>Editar mapas</i></li> <li>3. Ir a pestaña <i>Mapas</i>.</li> <li>4. Seleccionar la vista deseada.</li> </ol>



<b>CU-0107</b>	<b>Descargar mapa de GM</b>
Actores	Usuario.
Descripción	Descarga de un mapa de Google Maps.
Precondición	Conexión a Internet disponible y modo Google Maps seleccionado.
Postcondición	Almacenamiento de las imágenes de Google Maps y fichero asociado.
Flujo principal	<ol style="list-style-type: none"> <li>1. Elegir idioma.</li> <li>2. Seleccionar <i>Navegación libre</i> o <i>Editar mapa</i>.</li> <li>3. Ir a la pestaña <i>Mapas</i>.</li> <li>4. Seleccionar <i>Guardar mapa</i>.</li> <li>5. Navegar por el sistema de archivos, asignar un nombre y guardar.</li> </ol>

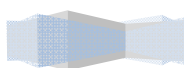
<b>CU-0117</b>	<b>Desplazar un mapa</b>
Actores	Usuario.
Descripción	Desplazarse por un mapa para visualizar su contenido.
Precondición	Tener un mapa cargado o conexión a Internet en modo Google Maps.
Postcondición	
Flujo principal	<ol style="list-style-type: none"> <li>1. Elegir idioma.</li> <li>2. Seleccionar <i>Navegación libre</i> o <i>Editar mapa</i>.</li> <li>3. Pinchar con el botón derecho del ratón sin soltar sobre el mapa.</li> <li>4. Desplazarse por el mapa y soltar el ratón</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>4.1 En <i>Navegación libre</i>, para volver a la posición actual, pinchar en <i>Volver</i>.</li> </ol>



<b>CU-0118</b>	<b>Centrar un mapa en una posición deseada</b>
Actores	Usuario.
Descripción	Colocar una localización seleccionada en el centro de la pantalla.
Precondición	Tener un mapa cargado o conexión a Internet en modo Google Maps.
Postcondición	
Flujo principal	<ol style="list-style-type: none"> <li>1. Elegir idioma.</li> <li>2. Seleccionar <i>Editar mapa</i>.</li> <li>3. Pinchar con el botón derecho del ratón sobre el mapa.</li> <li>4. Seleccionar <i>Centrar mapa</i></li> </ol>
Flujo alternativo	

## Pois

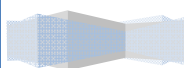
<b>CU-0200</b>	<b>Añadir POI</b>
Actores	Usuario.
Descripción	Marcar como punto de interés una localización en un mapa.
Precondición	Tener un mapa cargado o conexión a Internet en modo Google Maps.
Postcondición	Se muestra el POI sobre el mapa.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar <i>Navegación libre</i>.</li> <li>3. Ir a la pestaña POIs.</li> <li>4. Pinchar el botón <i>Añadir POI</i>.</li> <li>5. Marcar la posición del lugar en el mapa con el ratón.</li> <li>6. Rellenar la información del POI y aceptar.</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>4.1 En lugar de pinchar en el botón, se puede hacer click con el botón derecho sobre el mapa y elegir <i>Añadir POI</i>.</li> </ol>



CU-0201	Borrar POI
Actores	Usuario.
Descripción	Eliminar un punto de interés de la lista de POIs.
Precondición	Tener un mapa cargado o conexión a Internet en modo Google Maps y al menos un POI en la lista de puntos de interés.
Postcondición	Se elimina el POI de la lista.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar <i>Navegación libre</i>.</li> <li>3. Ir a la pestaña POIs.</li> <li>4. Seleccionar un POI de la lista.</li> <li>5. Seleccionar <i>Borrar POI</i></li> </ol>

CU-0202	Guardar lista de POIs
Actores	Usuario.
Descripción	Crear un fichero CSV con los puntos de interés marcados.
Precondición	Tener un mapa cargado o conexión a Internet en modo Google Maps y tener al menos un punto de interés.
Postcondición	Se muestra el POI sobre el mapa.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar <i>Navegación libre o Editar mapas</i>.</li> <li>2. Ir a la pestaña POIs.</li> <li>3. Pinchar en el botón <i>Guardar POIs</i></li> <li>4. Navegar por el sistema de archivos, asignar un nombre y guardar la lista de POIs con formato .csv</li> </ol>

CU-0203	Cargar archivo de POIs
Actores	Usuario.
Descripción	Cargar un fichero CSV que contenga una serie de puntos de interés.
Precondición	Tener un mapa cargado o conexión a Internet en modo Google Maps y un fichero de puntos de interés en formato CSV.
Postcondición	Se carga en la lista de POIs la información



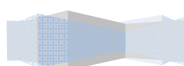


	almacenada en el fichero.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar <i>Navegación libre o Editar mapas</i>.</li> <li>2. Ir a la pestaña POIs.</li> <li>3. Pinchar en el botón <i>Cargar POIs</i></li> <li>4. Navegar por el sistema de archivos hasta encontrar la lista de POIs. Abrir el archivo</li> </ol>

<b>CU-0204</b>	<b>Editar la descripción de un POI</b>
Actores	Usuario.
Descripción	Editar la descripción de un punto de interés de la lista.
Precondición	Tener un mapa cargado o conexión a Internet en modo Google Maps y al menos un POI en la lista de puntos de interés.
Postcondición	Descripción del POI editado registrada.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar <i>Navegación libre o Editar Mapas</i>.</li> <li>3. Ir a la pestaña POIs.</li> <li>4. Seleccionar un POI de la lista.</li> <li>5. Editar la descripción en el cuadro de texto correspondiente</li> </ol>

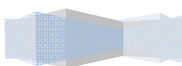
## Ayuda

<b>CU-0400</b>	<b>Consultar Ayuda</b>
Actores	Usuario.
Descripción	Consultar el manual de usuario
Flujo principal	<ol style="list-style-type: none"> <li>1. Elegir el idioma.</li> <li>2. Seleccionar <i>Ayuda</i>.</li> </ol>



## Calibración

<b>CU-0107</b>	<b>Calibrar un mapa asociado a una imagen</b>
Actores	Usuario.
Descripción	Creación de un mapa formato OziExplorer asociado a un archivo de imagen que represente un mapa.
Precondición	Disponer de un archivo de imagen de uno de los tipos permitidos y conocer las coordenadas de dos puntos o tener acceso a un dispositivo GPS.
Postcondición	Creación de un fichero con extensión .map en formato OziExplorer.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar <i>Calibración</i>.</li> <li>3. Seleccionar <i>Cargar imagen</i>.</li> <li>4. Navegar por el sistema de archivos hasta localizar la imagen deseada.</li> <li>5. Seleccionar el archivo de imagen.</li> <li>6. Seleccionar <i>Punto1</i>.</li> <li>7. Marcar en el mapa la posición del primer punto conocido.</li> <li>8. Escribir la latitud y la longitud del primer punto conocido.</li> <li>9. Seleccionar <i>Punto2</i>.</li> <li>10. Marcar en el mapa la posición del segundo punto conocido.</li> <li>11. Escribir la latitud y la longitud del segundo punto conocido.</li> <li>12. Seleccionar <i>Calibrar</i>.</li> <li>13. Asignar un nombre al fichero mapa generado. Navegar por el sistema de archivos hasta localizar la ubicación deseada.</li> <li>14. Aceptar y guardar el mapa.</li> </ol>
Flujo alternativo	<ol style="list-style-type: none"> <li>5.1. Si se intenta cargar un archivo no permitido, no se avanza.</li> <li>7.1. Si se desea modificar el lugar indicado, repetir el paso 7.</li> <li>8.1. Si se desea modificar el valor, repetir el paso 8.</li> <li>2.1. Si se desea utilizar las</li> </ol>

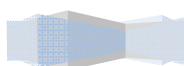


	<p>coordenadas suministradas por el GPS, seleccionar Usar <i>datos Gps</i>.</p> <p>10.1 Si se desea modificar el lugar señalado, repetir el paso 10.</p> <p>11.1 Si se desea modificar el valor, repetir el paso 11.</p> <p>11.2 Si se desea utilizar las coordenadas suministradas por el GPS, seleccionar Usar <i>datos Gps..</i></p>
--	---

## Configuración

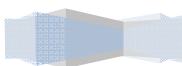
CU-0108	Configurar la utilización de mapas propios
Actores	Usuario.
Descripción	Establecer el uso de mapas propios a la hora de navegar.
Postcondición	Uso de mapas propios predeterminado.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar <i>Configuración</i>.</li> <li>3. Marcar la opción <i>Disco duro</i>.</li> <li>4. Seleccionar <i>Aceptar</i>.</li> </ol>
Flujo alternativo	2.1. Si se aborta la operación se vuelve al menú principal

CU-0109	Configuración de idioma
Actores	Usuario.
Descripción	Selección del lenguaje predeterminado
Postcondición	Menús y rótulos en el idioma seleccionado.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar <i>Configuración</i>.</li> <li>3. Elegir un idioma.</li> </ol>
Flujo alternativo	2.1. Si se aborta la operación se vuelve al menú principal.



CU-0110	Configuración del puerto serie
Actores	Usuario.
Descripción	Modificación del puerto serie utilizado para la conexión GPS.
Postcondición	Puerto de conexión con el dispositivo GPS modificado.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar <i>Configuración</i>.</li> <li>3. Editar el nuevo valor del <i>Puerto serie</i>.</li> </ol>
Flujo alternativo	2.1. Si se aborta la operación se vuelve al menú principal.

CU-0110	Configuración de la cuenta Google
Actores	Usuario.
Descripción	Modificación del valor por defecto de la cuenta Google.
Precondición	Disponer de una cuenta para el uso del API de Google Maps.
Postcondición	Uso de la nueva cuenta de Google.
Flujo principal	<ol style="list-style-type: none"> <li>1. Seleccionar idioma.</li> <li>2. Seleccionar <i>Configuración</i>.</li> <li>3. Editar el nuevo valor de la <i>Cuenta Google</i>.</li> <li>4. Seleccionar <i>Aceptar</i>.</li> </ol>
Flujo alternativo	2.1. Si se aborta la operación se vuelve al menú principal.



## VIABILIDAD Y ALCANCE

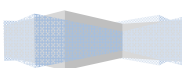
Ambos puntos han estado muy presentes a lo largo de este año durante el desarrollo del proyecto. A principio de curso nos marcamos unos objetivos con la supervisión y consejo de nuestro tutor, donde nos comprometimos a desarrollar una aplicación completa, vistosa y de utilidad para el usuario final. Siempre hemos tenido en mente estos objetivos durante el desarrollo para conseguir que *Geomantes<sup>PC</sup>* posea dichas características marcadas.

Al mismo tiempo hemos de reconocer que nos hubiera gustado abordar otras vías alternativas del proyecto para hacerlo aún más sólido y completo y que finalmente no hemos podido -en unas ocasiones- o sabido -en otras- llevar a cabo y hemos ido abandonando por resultarnos inviables. Unas veces la inexperiencia, otras la falta de tiempo o de recursos materiales han sido los principales motivos para descartar estas ramas que teníamos abiertas como posibles mejoras a añadir a la aplicación.

Una de ellas fue la incursión en el mundo de las nuevas plataformas para móviles, más concretamente en la plataforma *Android*. Buscando ampliar la compatibilidad de la aplicación *Geomantes* (versión para dispositivos móviles) desarrollada por unos compañeros el curso 2008/2009 de la que tomábamos el testigo, comenzamos a indagar en la compatibilidad de la aplicación con este nuevo sistema operativo desarrollado por Google por su compromiso con los estándares abiertos y por encontrarse a la vanguardia del desarrollo de software libre para este tipo de dispositivos.

Esta vía se vio frustrada tras varios meses de investigación y lentos avances en el desarrollo de la aplicación por uno de los integrantes del grupo. Los problemas derivados de la inexperiencia en la arquitectura de *Android* unidos al aislamiento en el que Alejandro se encontraba en su desarrollo -salvo por las reuniones semanales en las que poco o nada podíamos poner en común debido a las diferencias entre sus dificultades y las nuestras-, le hicieron plantearse hasta qué punto podría llegar siquiera a pronosticar cuanto tiempo le llevaría portar una aplicación grande y compleja como la de *Geomantes* si, tras varios meses dedicados a la tarea, no conseguía obtener avances significativos. Esta idea, junto a su firme convicción de poder ser útil a sus compañeros en la vía principal de desarrollo, hizo que declaráramos como **inviable** esta opción para este curso.

Otra de las vías alternativas de desarrollo que se vio frustrada fue la implementación de mapas vectoriales en la aplicación. No disponer de dispositivos que procesaran este tipo de ficheros ni acceso a la información interna del propio mapa complicaba mucho nuestro cometido. En una primera etapa, intentamos llevar a cabo la tarea de descifrar el contenido de estos mapas a base de ingeniería inversa probando con distintos

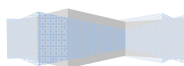


programas desarrollados para ello e interpretando los resultados para obtener la información base con la que crear nuestros propios mapas vectoriales. Tras numerosos intentos fallidos y mucho investigar en la materia, pudimos comprobar que una de las pocas fuentes fiables de información que nos quedaba para obtener mapas vectoriales se encontraba en el código libre y se llamaba [www.OpenStreetMap.org](http://www.OpenStreetMap.org). El método de almacenamiento de información usado por esta organización consiste en albergar en un fichero con la estructura propia de un xml un grafo de la situación donde vértices y aristas son, en realidad, cruces y vías que se interconectan para representar la realidad. Aunque, a primera vista, este sistema nos pareció acertado para llevar a cabo nuestro cometido, dándonos la posibilidad de implementar sobre esta estructura algoritmos de búsquedas de caminos óptimos sobre grafos para ofrecer al usuario el camino más corto entre un punto inicial y el destino, pronto pudimos comprobar la intratabilidad del problema debido al crecimiento exponencial de nodos según se amplía la zona a visualizar a un tamaño más o menos aceptable. Ni siquiera la página es capaz de proporcionar mapas de tamaños superiores a unos pocos cientos de metros mostrando el siguiente mensaje por respuesta: *Bad Request: You requested too many nodes (limit is 50000)*. Todo esto nos llevó a aparcarse momentáneamente esta idea debido a la capacidad de procesamiento que el problema requiere y del que los móviles carecen, haciendo intratable su cómputo y por tanto **inviable** en este tipo de plataformas.

Fue entonces cuando, decididamente, apostamos por trasladar la aplicación que recibíamos a un sistema más potente que pudiera afrontar mayores retos. Por ello, pensamos en crear una nueva versión que pudiera ser corrida sobre un ordenador portátil, para que sin perder la funcionalidad de la anterior versión ganara en capacidad de procesamiento y usabilidad ofreciendo un sistema de menús menos rígido y un área mayor de visualización del mapa.

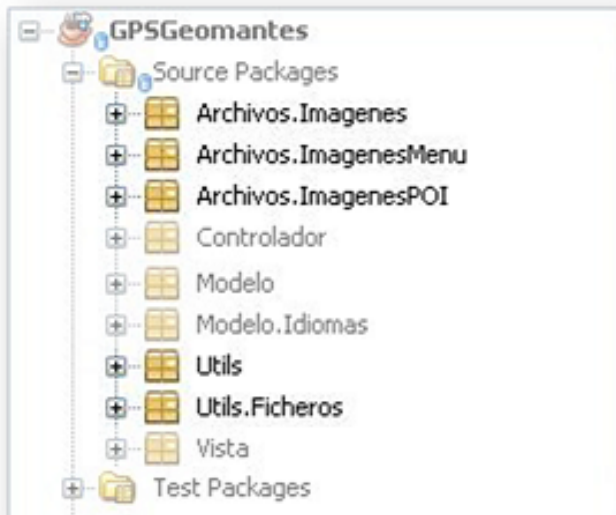
Sin embargo, desde un principio este nuevo reto resultó bastante más complejo de lo esperado, pues retomar una aplicación de tal magnitud en un estado tan avanzado puede dar vértigo si no se divide la comprensión del código en pequeñas tareas. A esta dificultad con la que contábamos se sumó otra de índole más grave y no esperada, pues numerosos módulos con los que contaba Geomantes en su versión desarrollada en *Java Micro Edition* resultaron no ser compatibles con la versión estándar de Java utilizada en aplicaciones para PC.

Después de todo y avanzando poco a poco, conseguimos ir solventando los problemas que encontramos y los nuevos que iban surgiendo, añadiendo las funcionalidades marcadas y puliendo una y otra vez la interfaz del usuario para finalmente poder afirmar que aquella aplicación que por Octubre recogimos se había convertido en una versión bastante digna de los propósitos que nos hicimos a principio de curso.



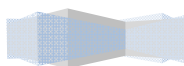
## DISEÑO

Hemos guiado el desarrollo del proyecto por los patrones de diseño *Singleton* y, principalmente, *Modelo-Vista-Controlador*, cuyo uso ha influido directamente sobre la estructura del código. Hemos intentado dividir y separar distintos fragmentos de código en un conjunto de paquetes y subpaquetes para lograr el mayor grado de modularidad posible.



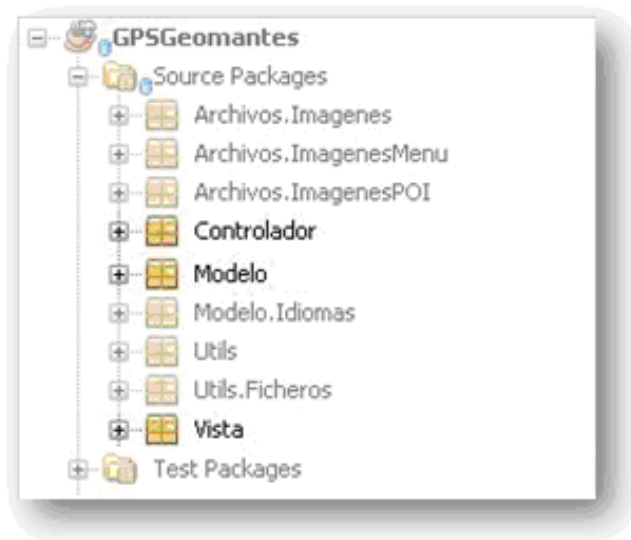
En adición a los ya mencionados Modelo, Vista y Controlador, hemos creados otros paquetes para agrupar aquellas clases que por su funcionalidad han sido encapsuladas conjuntamente.

Podemos observar dos familias distintas de paquetes auxiliares: los relativos a archivos de imágenes utilizados en las vistas o interfaces de usuario (agrupados bajo el término **Archivos**); y los que hacen referencia a clases y métodos auxiliares utilizados para la geolocalización (conversión de coordenadas geográficas y coordenadas UTM, cálculo de distancias relativas a dos puntos dados, etc.) y para la manipulación de ficheros de datos (lecturas y escrituras de rutas y puntos de interés). Todo esto se encuentra en el paquete **Utils**.

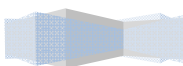


## Modelo-Vista-Controlador (MVC)

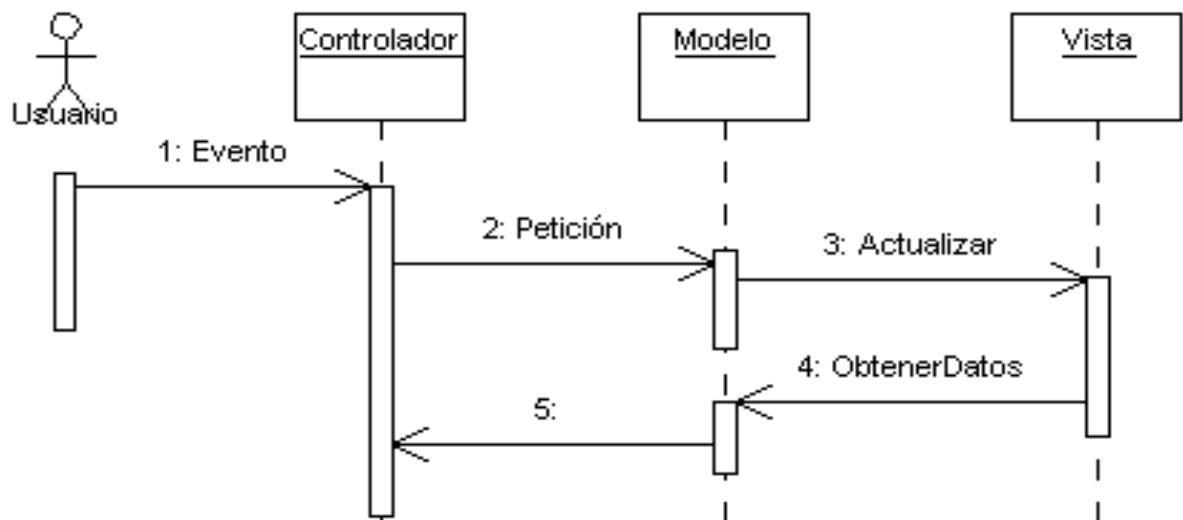
Este patrón de diseño más comúnmente utilizado en la construcción de páginas web, ha sido aplicado en nuestra aplicación por su cómoda implantación y fácil control sobre los distintos componentes. A continuación damos una pequeña introducción de los conceptos que definen este patrón aplicados a Geomantes<sup>PC</sup>:



- ✚ **Modelo:** agrupa el conjunto de datos dinámicos que utiliza la aplicación y que definen su estado. Se puede considerar como el almacén de valores suministrados por los distintos proveedores de datos (como las coordenadas capturadas del dispositivo GPS o los mapas descargados del servicio de Google Maps) junto con los propios parámetros definidos por la aplicación y el usuario que configuran el estado del navegador.
- ✚ **Vista:** forma de representar los datos que contiene el Modelo de manera gráfica. Hemos optado por crear un repertorio de vistas limpias y atractivas para facilitar su uso a todo tipo de usuarios.
- ✚ **Controlador:** es el módulo encargado del control y gestión de los datos y de seleccionar la vista adecuada en cada momento. De alguna manera es el cerebro que alberga toda la lógica necesaria para el correcto funcionamiento del sistema. Su estructura es compleja y abarca la gestión de todos los servicios que se hayan presentes en nuestra aplicación. Esta clase ha sido implementada bajo el patrón *singleton* para asegurar la existencia de una única instancia en todo momento.



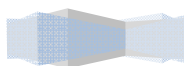


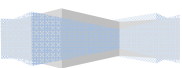
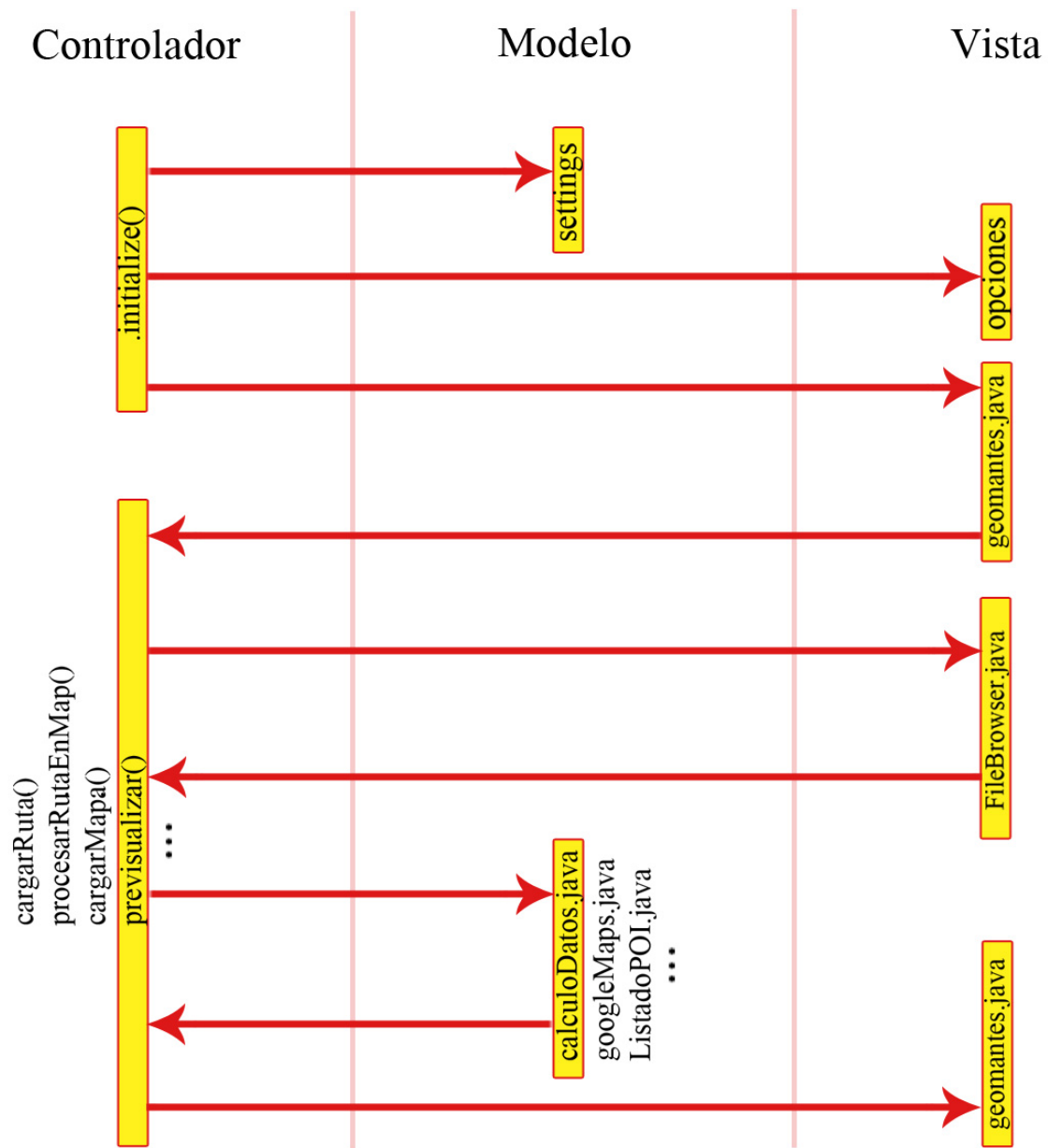


*Diagrama de interacción entre los tres módulos.*

Más adelante mostramos la aplicación práctica de estos conceptos y del diseño MVC, mediante un diagrama que ejemplifica uno de los usos que se pueden hacer de la aplicación.

Gracias al empleo de esta técnica hemos evitado la repetición de código, reutilizando aquellos módulos que tan sólo necesitaban ligeras modificaciones. Además, se fomenta la extensibilidad ya que, por ejemplo, en una aplicación como Geomantes<sup>PC</sup> a menudo hay que modificar y adaptar las vistas en función del tipo de plataforma en la que se ejecute. En este caso, basta con añadir o modificar ligeramente las clases necesarias de la Vista, y en menor medida, del controlador, quedando intacta la parte de datos del Modelo.





## Singleton

Este patrón de diseño asegura la existencia de una única instancia del objeto de la clase que lo implemente. El propósito es evitar la proliferación de objetos y

Singleton	
-	<u>singleton : Singleton</u>
-	Singleton()
+	<u>getInstance() : Singleton</u>

proporcionar un punto de acceso global a la instancia. Es habitual su empleo en aquellos casos en los que sólo tenga sentido una instancia de clase, como ocurre con el Controlador en nuestro caso. En la imagen de la izquierda está representada la estructura estándar de este patrón. Nosotros hemos implementado este esquema

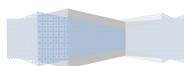
en nuestro código, con una constructora privada y accediendo a la instancia mediante un método estático, como se muestra a continuación:

```
private Controlador() {
    modoCalibracion = false;
    nombreRuta = "";
    nombreMapa = "";
    modoNavegacion = false; //Modo GPS por defecto
    modoGrabacion = false; //Por defecto no se está grabando
    mostrarRuta = true; //Por defecto se muestra la ruta
    mostrarPOI = true; //Por defecto se muestran los POI

    //Opciones iniciales
    modoMapas = true; //Modo GoogleMaps por defecto
    cuentaGoogle = "ABQIAAAAP7mVZDMT-Vpal7TCF1dcLhRk-Hwz9c_--F2C22uANDjvY";
    puertoSerie = 2;
    idioma = "es";
}

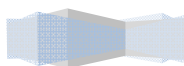
public static Controlador getInstance() {
    return ControladorHolder.instance;
}

private static class ControladorHolder {
    private final static Controlador instance = new Controlador();
}
```




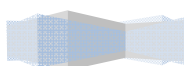
## Clase Controlador

Controlador
<p><i>Attributes</i></p> <pre> private boolean modoNavegacion private boolean modoGrabacion private boolean mostrarRuta private boolean mostrarPOI private JPanel displayMapa private Vector vectorRuta private Vector vectorRutaGeoPoint private int RUTA_MAX_SIZE = 50 private String nombreMapa private String nombreRuta private int anchoPantalla private int altoPantalla private boolean modoCalibracion private boolean modoMapas private int puertoSerie private String idioma private String cuentaGoogle private String tamano </pre>
<p><i>Operations</i></p> <pre> private Controlador( ) public Controlador getInstance( ) public Controlador getControlador( ) public void muestraAlerta( String mensaje ) public JPanel getDisplayMapa( ) public void setDisplayMapa( JPanel d ) public int getAnchoPantalla( ) public int getAltoPantalla( ) public void setAnchoPantalla( int ancho ) public void setAltoPantalla( int alto ) public String getIdioma( ) public void setIdioma( String i ) public String getTamano( ) public void setTamano( String t ) public int getPuertoSerie( ) public void setPuertoSerie( int port ) public String getCuentaGoogle( ) public void setCuentaGoogle( String r ) public boolean getModoMapas( ) public void setModoMapas( boolean b ) public void setMostrarPOI( boolean b ) public boolean getMostrarPOI( ) public void setListadoPOI( ListadoPOI p ) public ListadoPOI getListadoPOI( ) public void redibujarPOI( ) public void repaintGeomantes( FicheroDatos infoLinea, int datosEntrada[0..*], double velocidad, double distancia ) public Geomantes getGeomantes( ) public void iniciarGeomantes( ) public void reiniciarGeomantes( ) public void calibracion( GUI_Calibracion parent ) public boolean getModoCalibracion( ) public Calibracion getCalibracion( ) public boolean guardarRuta( Vector vectorR ) public boolean cargarRuta( ) public void procesarRutaEnMapa( ) private int reduceRuta( int contadorMax, int division ) public void redibujarRuta( Vector rutaLatLong ) public Vector getVectorRuta( ) public String getNombreRuta( ) public void anadeVectorRuta( int p[0..*]) public void deshacerMarcarRuta( ) public void iniciaRutaMarcada( ) public void setModoGrabacion( boolean b ) public boolean getModoGrabacion( ) public void setMostrarRuta( boolean b ) public boolean getMostrarRuta( ) public void setLocation( GPSPosition loc ) public boolean getModoNavegacion( ) public void setModoNavegacion( boolean b ) public void guardarMapaGoogle( int n, ElementoGmaps matrizGmaps[0..*,0..*], GoogleMaps googlemaps, int anchoP, int altoP ) public boolean cargarMapa( ) public String getNombreMapa( ) </pre>



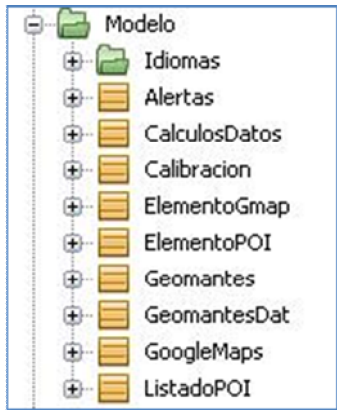
## Clase Geomantes

 <b>Geomantes</b>
<p><i>Attributes</i></p> <pre> private int datosEntrada[0..*] private double velocidad private double distancia private double longitudAntigua private double longitudActual private double latitudAntigua private double latitudActual private double escala private int nivelZoom private String mapType private String tipoVel private int despX private int despY private int ancho private int alto private Color colorRuta1 private Color colorRuta2 private Color colorRuta3 </pre>
<p><i>Operations</i></p> <pre> public Geomantes( ) public void setColor1( Color a ) public void setColor2( Color a ) public void setColor3( Color a ) public void setCoordenadas( FicheroDatos c ) public FicheroDatos getCoordenadas( ) public void setInfoFichero( FicheroDatos infolinea, int datosEntrada[0..*], double velocidad, double distancia ) private boolean estaCompleta( ElementoGmaps matriz[0..*,0..*] ) private void inicializarMemoria( ) private void paintMapasMemoria( Graphics g ) public void inicializarGmaps( ) private void paintMapasGoogle( Graphics g ) public void guardarMapas( ) private void desplazaMapasGmaps( ) private void paintNotFound( Graphics g ) private void paintRuta( Graphics g2 ) private void paintPuntosInteres( Graphics g2 ) public void paintBorder( Component c, Graphics g, int x, int y, int width, int height ) public Insets getBorderInsets( Component c ) public boolean isBorderOpaque( ) public void setSumaLatLong( double lat, double longi ) public int getNivelZoom( ) public void setNivelZoom( int nivelZoom ) public void setEscala( double e ) public double getEscala( ) public double getLatitud( ) public double getLongitud( ) public double getVelocidad( ) public double getDistancia( ) public String getMapType( ) public void setMapType( String s ) private void inicializarValores( ) private Image dameImagenPOI( ElementoPOI poi_element ) public ElementoGmaps[0..*,0..*] getMapaActual( ) public void limpiaMatrizMemoria( ) </pre>



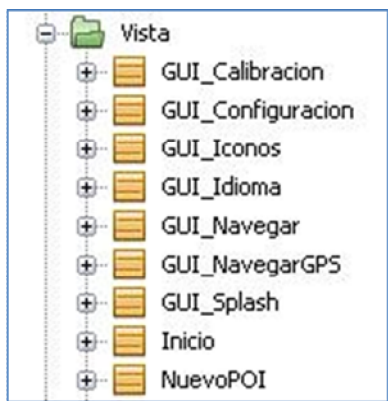
## Paquete Modelo

Contiene las siguientes clases:



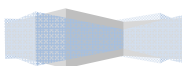
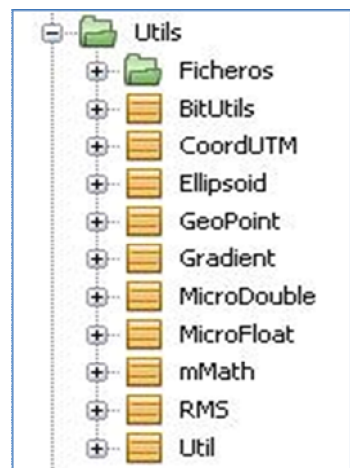
## Paquete Vista

Contiene las siguientes clases:



## Paquete Utils

Contiene las siguientes clases:



## PRUEBAS

El desarrollo de una aplicación es un proceso interactivo en el que a medida que se van implementando nuevas funcionalidades se comprueba su correcto funcionamiento. Sin embargo, es lógico que en un proyecto de estas dimensiones se produzcan conflictos cuando se acoplan distintos módulos. Es por ello, que pese a que las pruebas han estado presentes en cada una de las etapas de desarrollo, hemos reservado el tramo final para concentrar todos nuestros esfuerzos en aras de conseguir un software fiable y robusto.

## CALIDAD DEL SOFTWARE

Si atendemos a la definición dada por Roger S. Pressman, ingeniero del software de gran prestigio y miembro editor de IEEE, la calidad del software viene definida por:

*“Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente”*

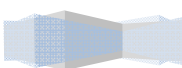
R. S. Pressman (1992)

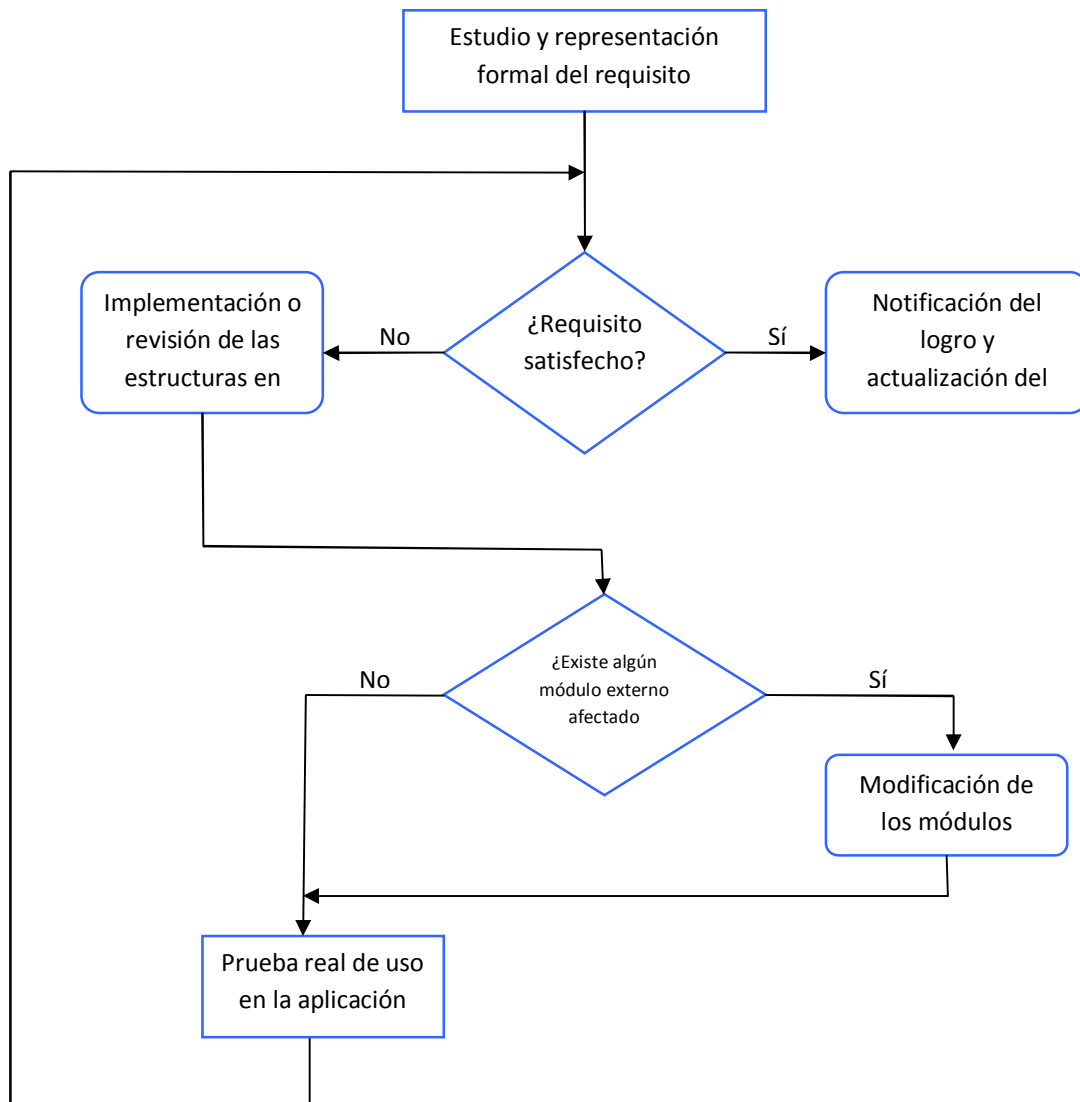
Los estándares de desarrollo **ISO-9000**, **ISO-9001** e **ISO-8402**, siendo este último un complemento de los anteriores para normalizar la terminología y precisar los conceptos, establecen su propia definición de calidad del software como sigue:

*“El conjunto de características de una entidad que le confieren su aptitud para satisfacer las necesidades expresadas y las implícitas”*

Interpretando estas definiciones observamos que para alcanzar un software de calidad se deben establecer unos procedimientos y técnicas de desarrollo que han de ser empleados de forma sistemática a lo largo de todo el ciclo de vida.

En nuestro trabajo hemos llevado a la práctica estos conceptos aplicando el siguiente esquema para cada uno de los requisitos exigidos al programa:

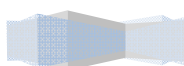




Gracias a este esquema hemos asegurado en la medida de lo posible un desarrollo consistente y fiable, teniendo presente por un lado los requisitos funcionales y por otro el código ya implementado hasta el momento.

Sin embargo, aunque estas medidas son necesarias para garantizar las bases y construir un software de calidad, no son suficientes al tratarse de un gran número de módulos trabajando sobre el mismo conjunto de datos. Así, hemos planificado un periodo de pruebas en el cuál cada integrante ha colaborado ha incrementar la fiabilidad poniendo a prueba los casos de uso, estresando a la aplicación y asegurando la corrección en todos los escenarios posibles.

Durante este periodo se ha creado una tabla compartida de tareas pendientes y se ha habilitado el acceso simultaneo a este mediante el servicio de Google docs.





Google docs Tareas pendientes

File Edit View Insert Format Form Tools Help No other users viewing.

Formula:

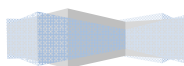
	A	B	C	D	E	F
1						
2	Nº Ref.	Descripción de la tarea	Nivel de prioridad	Estado	Asignado a	Comentarios
3	1	Permitir desplazamiento fuera de los límites del mapa	Alta	En proceso	Alejandro	Falla en los límites superior e izquierdo
4	2	Pintar rutas continua la ruta almacenada al grabar ruta	Media	Pendiente	Javi	
5	3	Navegación libre, falla cargar mapa	Media	Hecho	Alejandro	Revisar el método cargarMapa()
6	4	La descripción de los POIs no se debe poder editar cuando no hay uno seleccionado.	Baja	Pendiente	Alejandro	
7	5	Cuando pinchas un POI que no está en los mapas, en modo Google Maps, no carga los mapas	Alta	En proceso	Alfonso	Similar al caso de desplazarse
8	6	Si al cargar un mapa metes cualquier cosa que no sea un .map, falla	Baja	Pendiente	Javi	Comprobar extensión
9	7	Cargar fichero de puntos de interés en formato TomTom	Media	Hecho	Javi	
10	8	Salta excepción al navegar con GPS en modo Google Maps	Alta	En proceso	Alfonso	Problema del posicionamiento inicial

Hay que destacar que esta utilidad ha resultado de gran ayuda durante el último periodo y que no se descarta su uso para el mantenimiento y revisión de la aplicación.

## EJEMPLOS

Hemos provisto el proyecto de diversos ejemplos de rutas, puntos de interés y un archivo .map de la facultad de Informática de la Universidad Complutense de Madrid, para que el usuario pueda comprobar las vistas y funcionalidades que la aplicación ofrece de forma rápida y sencilla.

Puede encontrar los ejemplos dentro de la carpeta “Ejemplos” contenida en el archivador del proyecto.



# MANUAL DE USUARIO

## ¿Qué es Geomantes<sup>PC</sup>?

*Geomantes*<sup>PC</sup> es una aplicación Java pensada para ofrecer al usuario, tanto una navegación GPS sencilla y amigable sobre su PC, como la posibilidad de visualizar y editar de mapas a través de una interfaz gráfica intuitiva, fácil de usar y con numerosas posibilidades. Esta última funcionalidad está especialmente pensada para aquellos usuarios que no dispongan de dispositivo GPS y por ello se ha hecho un gran esfuerzo en dotar de un gran número de servicios a la pre-visualización de mapas.

Para ello hemos aprovechado las principales ventajas que poseen este tipo de dispositivos frente a los dispositivos móviles: la **capacidad de procesamiento** y el **tamaño de la ventana de navegación**.

Inicialmente fue pensada para ser usada sobre *laptops* (ordenadores portátiles) pero el mercado emergente de los *Tablet PCs* resulta mucho más atractivo para esta aplicación ya que mejora su usabilidad de manera increíble al satisfacer con un solo dispositivo todos los requisitos software y hardware que *Geomantes*<sup>PC</sup> necesita para su funcionamiento.

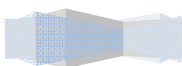
## Requisitos Hardware

### Dispositivo GPS

Si desea usar la funcionalidad que ofrece el menú *Navegación Libre GPS* es esencial que disponga de este dispositivo para que pueda ser geo-posicionado. En caso de no disponer de este elemento puede previsualizar el mapa de su situación si conoce sus coordenadas geográficas y dispone de conexión a internet.

### Bluetooth

Es necesario que el dispositivo GPS y el equipo donde va a correr la aplicación puedan comunicarse de alguna forma. Para ello hemos optado por una comunicación vía Bluetooth. El dispositivo móvil Nokia5800 dispone de ambas tecnologías haciendo viable la comunicación GPS-Computador. Hay que notar que el laptop también debe disponer de dispositivo Bluetooth para que pueda recibir los datos GPS que el móvil le manda. Hoy en día es bastante común que este tipo de PCs posean esta tecnología. En el caso de poseer un *Tablet PC* con GPS integrado este último elemento no será necesario al encontrarse ambos dispositivos en la misma máquina.



## Conexión a Internet

Si no dispone de mapas propios es necesario que su equipo tenga acceso a internet. Activando el modo *GoogleMaps* éstos serán descargados automáticamente desde <http://maps.google.com> siempre que se encuentre dentro de una zona con acceso a red y disponga del hardware necesario para conectarse a ella. Por tanto, necesita que su equipo posea un dispositivo receptor de señal Wifi o bien de cable de red.

## Requisitos Software

### Sistema Operativo

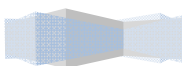
Al ser una aplicación desarrollada en Java EE puede ser ejecutada en cualquier sistema operativo que soporte JVM ( *Java Virtual Machine* o máquina virtual de Java)

### ExtGPS 0.31

Aplicación utilizada para el envío continuo de datos GPS desde el dispositivo móvil al PC donde se vaya a ejecutar *Geomantes*<sup>PC</sup>.

### Archivos externos

En el caso de necesitar la comunicación vía Bluetooth entre el dispositivo dotado de GPS y el ordenador personal, deberá ubicar la biblioteca *win32com.dll* y el archivo de configuración *javax.comm.properties* en este último para conseguir el correcto funcionamiento de Geomantes. Estos archivos los puede encontrar en el propio archivador de la aplicación. Si por el contrario, la máquina sobre la que va a ejecutar el programa dispone de GPS integrado, no necesita hacer uso de los archivos anteriormente mencionados. Además, deberá instalar Java Communication Api mediante la librería *comm.jar* incluida en el proyecto.



## Instalación

1. Extraiga el contenido del archivador *GeomantesPC* en la ubicación que desee.
2. Copie si es necesario los archivos *comm.jar*, *javax.comm.properties* y en la siguiente ruta (si no está seguro de si debe instalar estos archivos en su ordenador consulte el tercer apartado de la sección Requisitos Software):

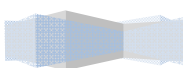
- ✓ *comm.jar*: PATH\_JRE\lib\ext
- ✓ *javax.comm.properties*: PATH\_JRE\lib\
- ✓ Si está en Windows:
  - i. *win32com.dll*: PATH\_JRE\bin\
- ✓ Si está en Linux:
  - i. *libLinuxSerialParallel.so*: usr\bin\

*(El path usual del JRE en Windows suele ser "C:\Archivos de programa\Java\jre6\")*

**Nota:** la versión de JRE debe coincidir con la del JDK utilizado en la compilación para asegurar el buen funcionamiento de la aplicación.

[ Version JDK usada : 1.6.0]

3. En caso no de querer navegar a través del GPS del dispositivo, vaya directamente al paso 5.
4. Para la instalación de la aplicación ExtGps, siga las siguientes instrucciones.
  - ✓ Copie la aplicación ExtGps en el dispositivo móvil.
  - ✓ Instale la aplicación en el dispositivo.
  - ✓ Ejecute la aplicación:
    - i. Acepte activar el Bluetooth de su dispositivo móvil
    - ii. Acepte activar el Gps de su dispositivo móvil.
    - iii. Conecte el móvil al ordenador con el software del Bluetooth que le haya proporcionado el fabricante.
5. Haga doble click sobre el archivo *GeomantesPC.jar* para disfrutar de la aplicación.



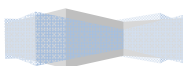
## Sistema de Menús

### Menú Principal

Muestra las principales funcionalidades que ofrece la aplicación Geomantes. Su apariencia es la siguiente:

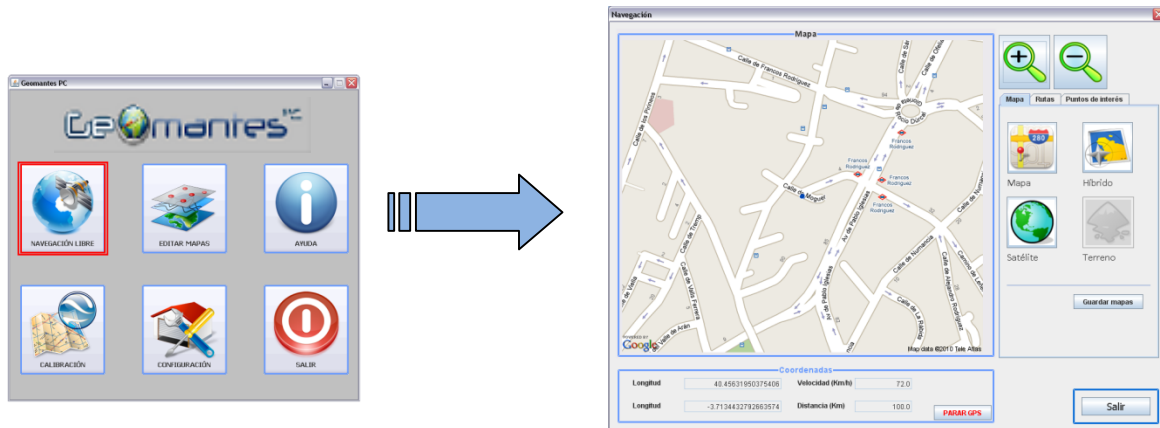


A continuacion se detalla el flujo de pantallas que desencadena la selección de los distintos botones de este menú.



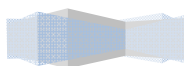
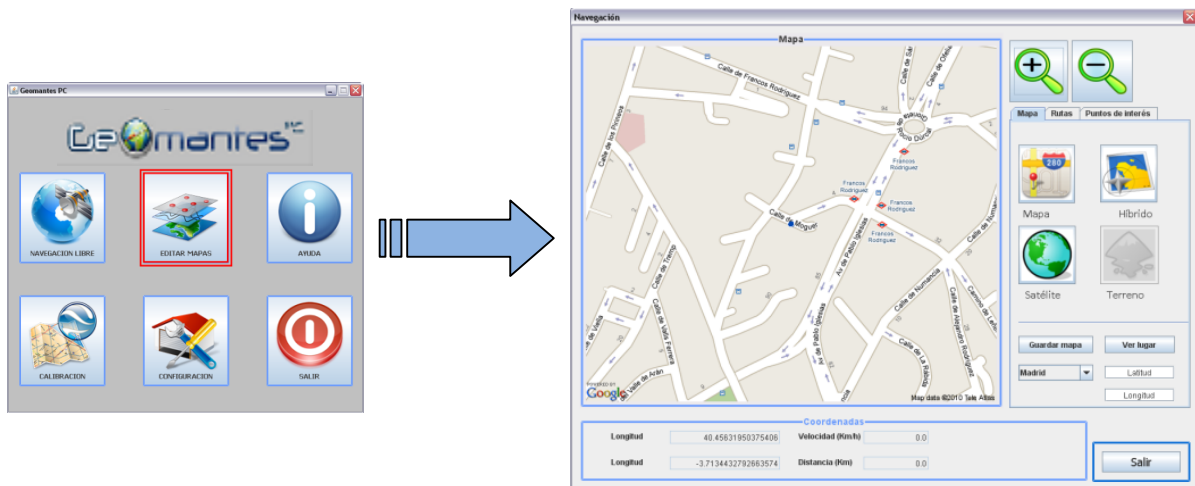
## Botón Navegación Libre

Al pulsar este botón se muestra por pantalla el panel de navegación. En este nuevo menú el usuario puede ver tanto su geolocalización en el mapa como usar las diferentes funcionalidades que este menú ofrece.



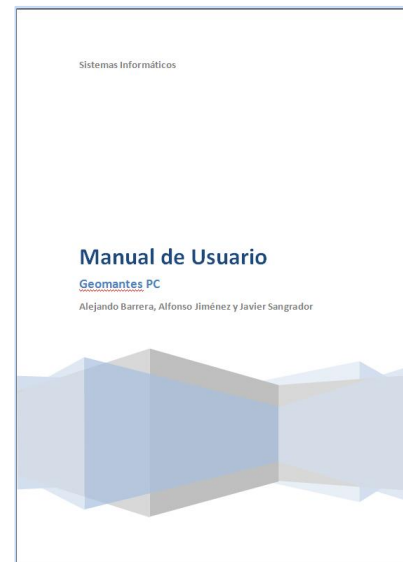
## Botón Editar Mapas

Al pulsar este botón se muestra por pantalla el panel de edición de mapas. En este panel el usuario puede añadir rutas, puntos de interés o visualizar el mapa desde distintas vistas.



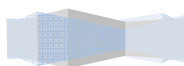
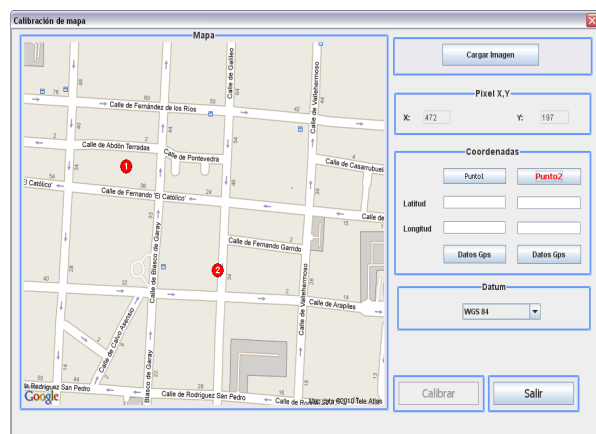
## Botón Ayuda

Al pulsar el botón de ayuda se muestra al usuario este manual de uso de la aplicación para navegación GPS en su versión para PC *Geomantes*.



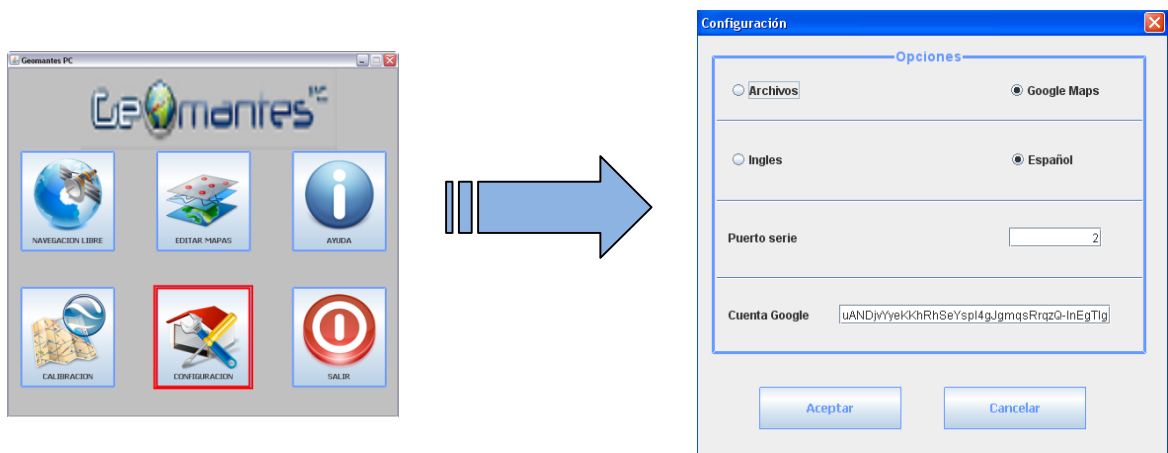
## Botón Calibración

Este botón muestra al usuario el panel para calibración de mapas. En este nuevo menú el usuario puede cargar imágenes de mapas que tenga almacenadas en su PC y convertirlas en mapas navegables en Geomantes.



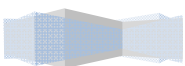
### Botón Configuración

Pulsando este botón muestra al usuario el menú de configuración donde puede elegir sus preferencias a la hora de usar *Geomantes*.



### Boton Salir

Finaliza la ejecución del programa





## Menú Configuración

**Configuración**

**Opciones**

☐ Archivos ☒ Google Maps

☐ Ingles ☒ Español

Puerto serie

Cuenta Google

Aceptar Cancelar

Desde este menú el usuario puede configurar las opciones que quiere utilizar en cada momento. Empezando desde arriba, puede elegir entre las siguientes opciones:

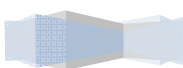
### Opciones sobre la obtención de mapas

**Opciones**

☐ Archivos ☒ Google Maps

La opción por defecto es Google Maps. Marcar esta casilla implica que tanto los mapas que se utilizan durante la navegación como los que se utilizan en la edición, provienen directamente de la página de Google Maps. Por tanto el usuario no necesita tener previamente almacenado en el equipo ningún mapa.

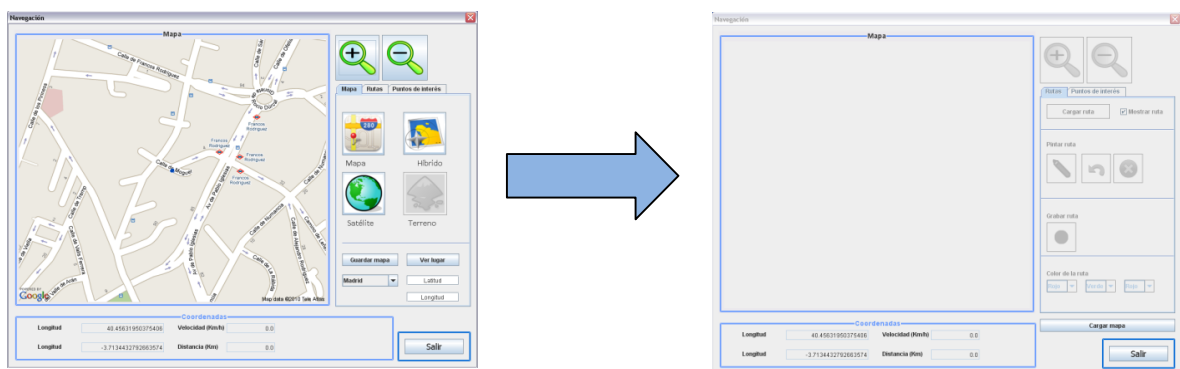
Esta modalidad necesita que el equipo tenga acceso a internet pues los mapas utilizados para la navegación son descargados en tiempo real desde la página <http://maps.google.es>



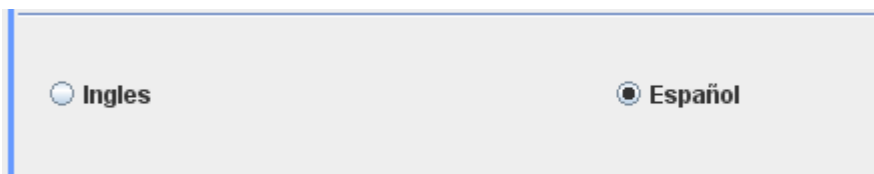
Habilitando la opción Archivos, el usuario puede cargar sus propios mapas en la aplicación. Al marcar esta casilla se deshabilitará la de “Google Maps” ya que ambas son alternativas excluyentes sobre la obtención de los mapas que se utilizan durante la ejecución de la aplicación.

Esta opción es especialmente útil cuando se encuentra en lugares sin acceso a internet o quiere utilizar un mapa diferente al que ofrece Google Maps.

Al tener habilitada esta opción la apariencia de algunas pantallas cambia para ofrecer funciones que en el modo “Google Maps” no son necesarias y viceversa.

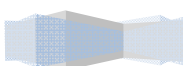


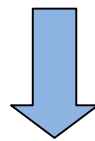
## Opciones de Idioma



Mediante esta opción el usuario puede elegir el idioma deseado. La opción que aparece marcada por defecto depende de la seleccionada al iniciar el programa en el cuadro de elección de idioma. En cualquier caso, el usuario siempre puede volver aquí y reconsiderar su elección seleccionando una casilla u otra.

El cambio de idioma da lugar a que los diferentes elementos de la aplicación muestren el texto en inglés o en español como se puede apreciar a continuación.





### Puerto Serie

<b>Puerto serie</b>	<input type="text" value="2"/>
---------------------	--------------------------------

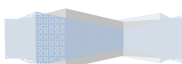
Desde aquí puede configurar el puerto serie por el que se comunicarán el dispositivo GPS y su PC.

### Cuenta de Google

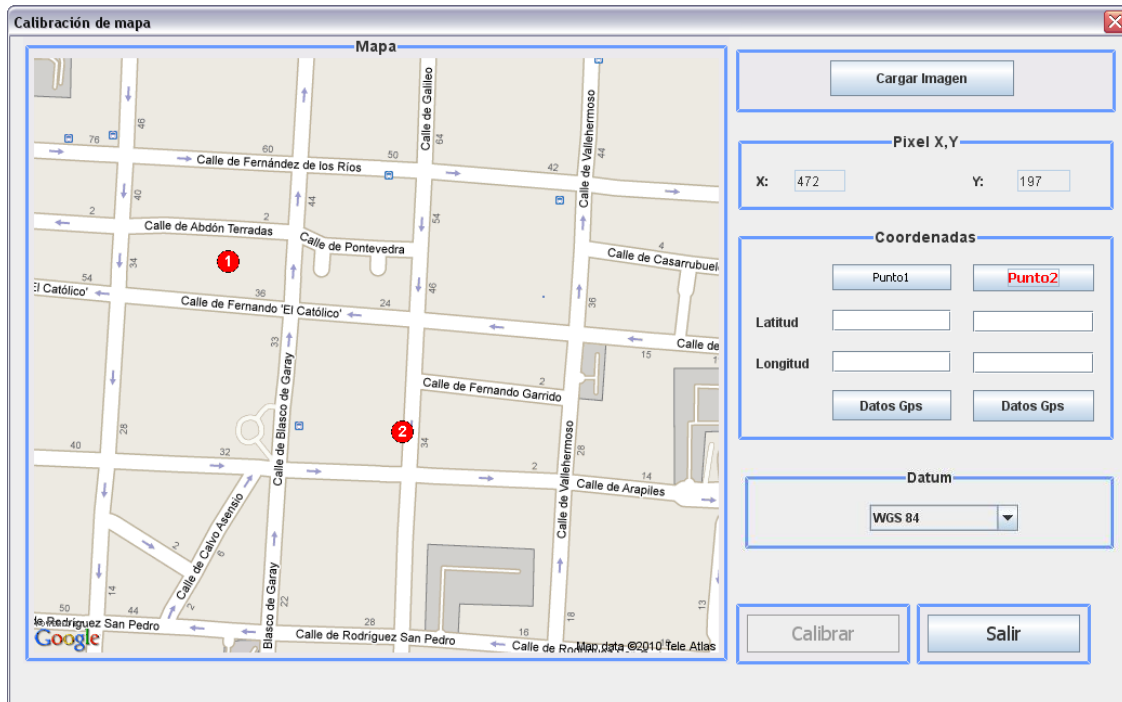
<b>Cuenta Google</b>	<input type="text" value="uANDjvYyeKKhRhSeYspl4gJgmqsRrQzQ-InEgTlg"/>
----------------------	---

Este código de cuenta que aparece por defecto es el que le permite poder navegar en modo google maps. Si usted posee su propia cuenta de google maps puede cambiar este código por el suyo.

Para obtener su propia cuenta de google maps pinche [aquí](#).



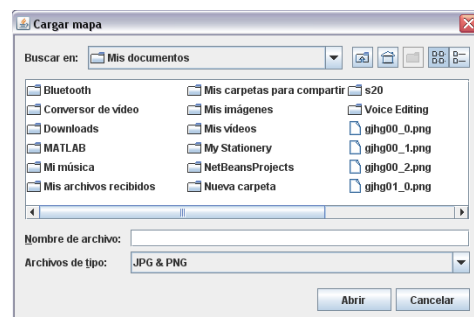
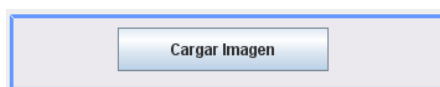
## Menú Calibración



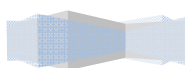
Este menú le ofrece la posibilidad de cargar imágenes que tenga almacenadas en su propio PC y transformarlas en mapas navegables en la aplicación. Para ello deberá identificar dos puntos del mapa y proporcionar sus coordenadas (latitud y longitud), o bien, si dispone de dispositivo GPS, dejar que sea éste quien proporcione las coordenadas necesarias.

A continuación se muestran los diferentes elementos de este menú y su cometido:

### Cargar Imagen



Pulse este botón para elegir la imagen cartográfica que desee calibrar. Al pulsar este botón verá en pantalla un cuadro de diálogo que le pedirá que elija una de entre las imágenes que se encuentran en su ordenador. Tenga en cuenta que los formatos de imagen soportados por la aplicación son PNG y JPG



## Pixel X,Y

Meramente informativa, muestra en cada momento el pixel de la imagen sobre el que se encuentra el puntero del ratón.

A dialog box titled "Pixel X,Y" with two input fields. The "X:" field contains the value "472" and the "Y:" field contains the value "197".

## Coordenadas

En este cuadro se han de introducir las coordenadas correspondientes a los puntos pulsados sobre la imagen para que el algoritmo de calibración pueda transformar la imagen en mapa.

A dialog box titled "Coordenadas" with two columns of controls. The top row has buttons "Punto1" and "Punto2" (the latter is red). Below are input fields for "Latitud" and "Longitud" for both points. At the bottom are buttons labeled "Datos Gps" for each point.

Los botones Punto1 y Punto2 sirven para elegir el punto a marcar sobre la imagen en un instante determinado. Los cuadros de texto en blanco han de ser rellenados con la longitud y latitud de los puntos respectivos que haya marcado sobre la imagen.

Si dispone de GPS puede pulsar los botones Usar datos Gps para que los cuadros se rellenen automáticamente con los datos obtenidos por el dispositivo. Tenga en cuenta que los puntos que aporte siempre deben encontrarse a una distancia mínima en la realidad de **100 metros** para que el algoritmo que calibra la imagen pueda funcionar con precisión.

## Datum

Puede elegir entre distintos modelos matemáticos de la Tierra. Por defecto aparece el *WGS 84* por ser el más comúnmente utilizado

A dialog box titled "Datum" with a dropdown menu currently showing "WGS 84".

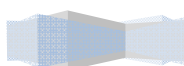
## Calibrar

A button with the text "Calibrar" in red.

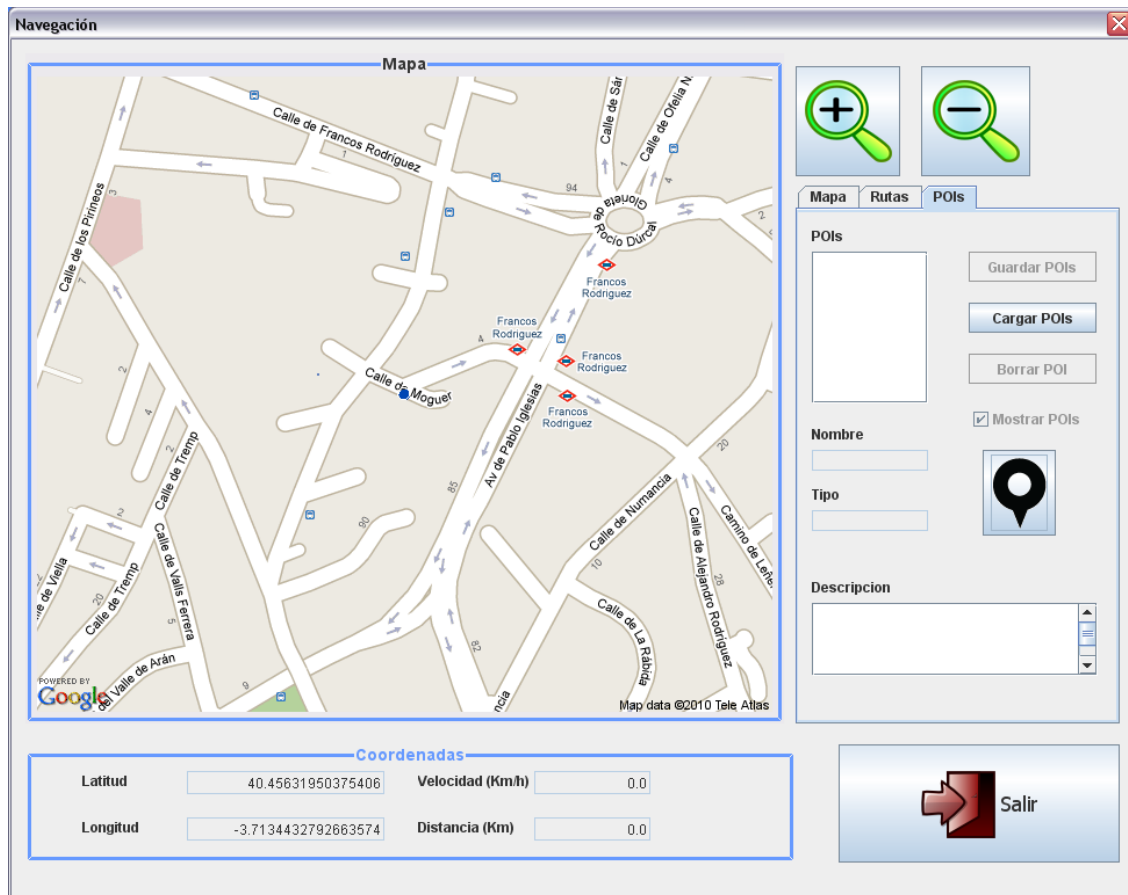


A "Guardar mapa" (Save map) dialog box. It shows a file explorer view of "Mis documentos" with various folders and files. At the bottom, there are fields for "Nombre de archivo:" and "Archivos de tipo:" (set to \*.map), and "Guardar" and "Cancelar" buttons.

Una vez estén todos los datos proporcionados el botón calibrar aparecerá habilitado. Pulse este botón si desea que se genere un mapa calibrado a partir de la imagen y los puntos suministrados. Aparecerá un cuadro de dialogo pidiéndole que especifique la ruta donde desee guardar el mapa generado ( el mapa tiene extensión .map)



## Menú Editar Mapas



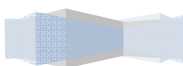
Este menú permite al usuario personalizar sus mapas previamente a la navegación para dotarlos de mayor funcionalidad. Entre los elementos que el usuario puede añadir se encuentran los puntos de interés que considere importantes en su mapa o la creación de rutas.

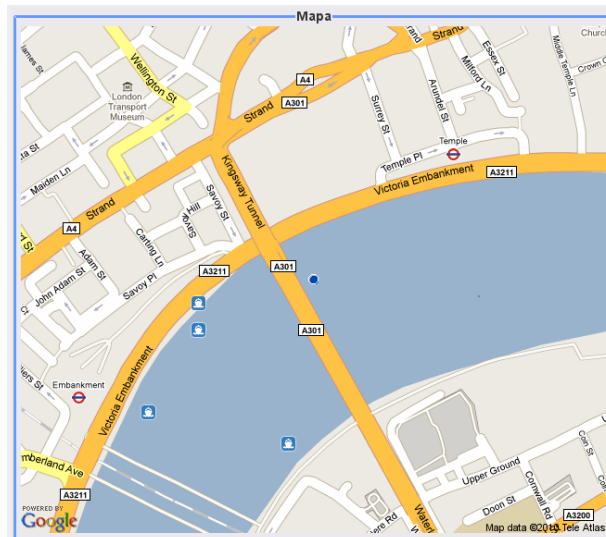
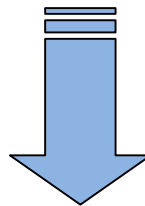
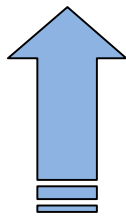
Además, si se encuentra en modo GoogleMaps podrá elegir entre distintos tipos de vistas para el mapa, o visualizar directamente alguna de las ciudades más importantes o incluso guardar el mapa en su propio PC.

A continuación se detallan los elementos mas importantes de esta pantalla:

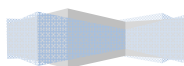
### Zoom

Estos botones permiten al usuario ampliar o reducir la imagen en función de sus necesidades. Si se encuentra en modo GoogleMaps, al pulsar el botón **Zoom+** se mostrará por pantalla un mapa más detallado de la zona en la que se encuentra o una perspectiva mas amplia si pulsa **Zoom-**.





En caso de estar visualizando un mapa propio, el zoom simplemente aumenta o reduce el tamaño de la imagen.





## Cuadro de Coordenadas

Coordenadas			
Latitud	<input type="text" value="48.860666538371596"/>	Velocidad (Km/h)	<input type="text" value="72.0"/>
Longitud	<input type="text" value="2.3343030168163614"/>	Distancia (Km)	<input type="text" value="11040.0"/>

El cuadro de coordenadas ofrece al usuario información sobre su situación en cada momento. Los campos **Latitud** y **Longitud** muestran coordenadas geográficas del usuario en cada instante posicionándolo unívocamente en el sistema de coordenadas terrestre. El campo **Velocidad** muestra en Kilometros por hora la velocidad actual del usuario y el campo **Distancia** informa sobre los Kilometros recorridos por el usuario desde el comienzo de la edición del mapa.

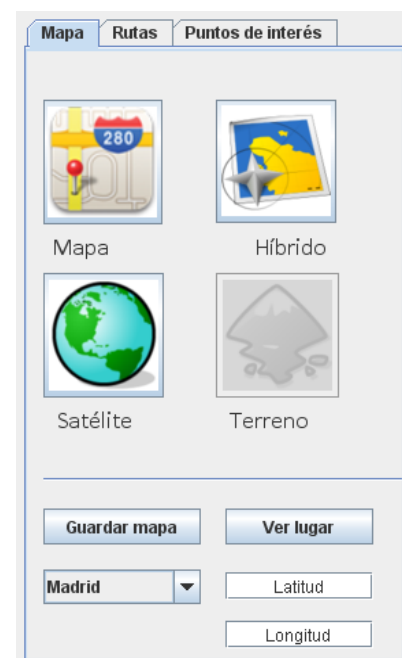
## Panel de funcionalidades

Este panel permite al usuario trabajar sobre el mapa ofreciéndole la posibilidad de añadirle rutas, puntos de interés o visualizarlo desde distintos tipos de vistas. Contiene 3 pestañas que se detallan a continuación.

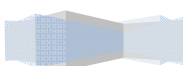
### Pestaña Mapa

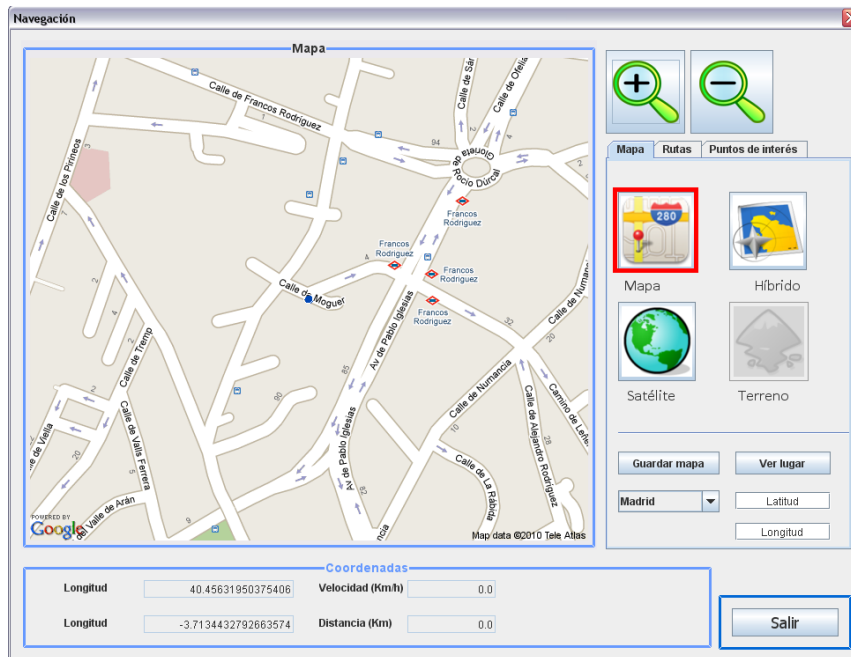
Esta pestaña solo es accesible en modo Google Maps. Contiene una primera sección con 4 botones que muestran 4 tipos de vista distintos. La sección inferior ofrece la posibilidad de situarse automáticamente en alguna de las ciudades mas importantes de Europa.

A continuación se muestra la apariencia de las diferentes vistas:

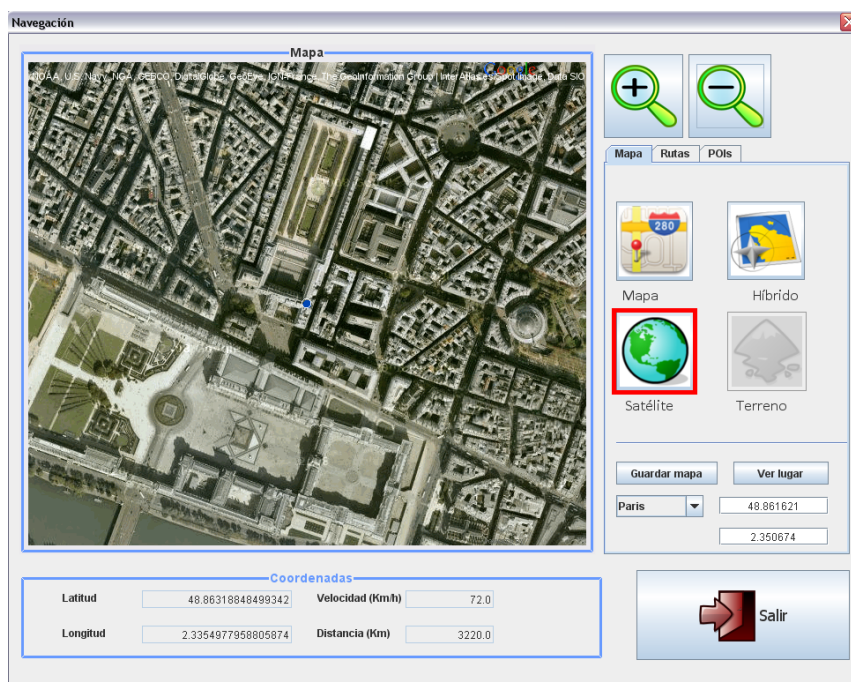


Mapa: Plano que contiene información sobre las calles y demás elementos importantes de la zona

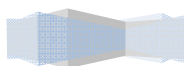


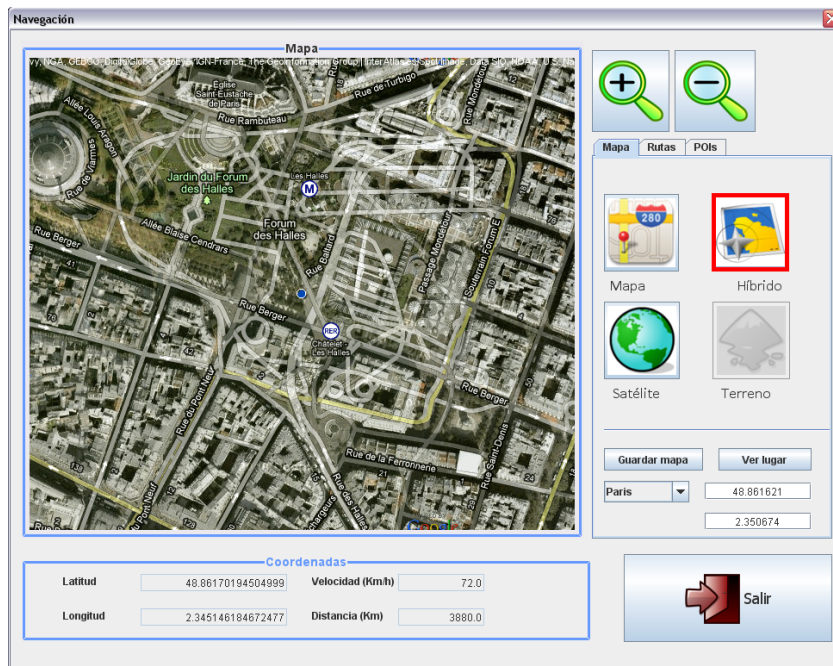


**Satélite:** Muestra por pantalla una imagen de la zona sin ningún tipo de información adicional.

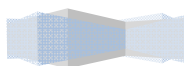
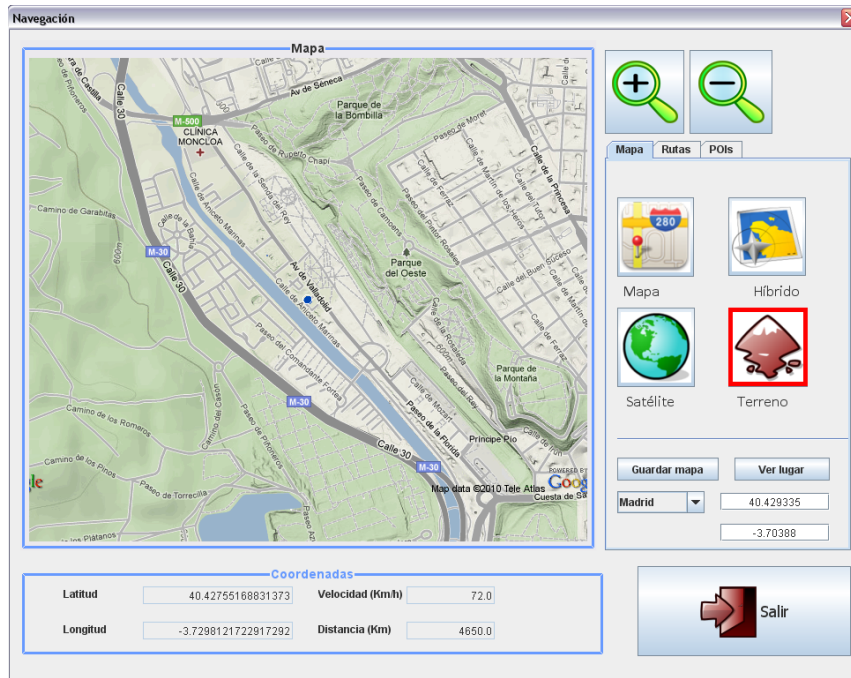


**Híbrido:** Ofrece una imagen satelital de la zona con información añadida sobre las calles, monumentos y demás elementos a destacar.





Terreno: muestra una imagen del relieve físico de la zona en la que aparece terreno y vegetación.



### Eleccion de Ciudades:

El menú desplegable que contiene el nombre de las ciudades le muestra los lugares entre los que puede elegir. Si pulsa en alguna de ellas se cargarán sus coordenadas geográficas en los cuadros de texto *latitud* y *longitud*. También puede introducir directamente en estos cuadros el valor que desee. Pulse el botón *Ver lugar* para que se muestre en el mapa el lugar especificado. Puede guardar el mapa pulsando el botón *Guardar Mapa*. Se le mostrará al usuario un cuadro de diálogo pidiéndole que indique el lugar donde desea guardar el mapa. Éste se compone de un archivo .map y una matriz de imágenes.




### **Pestaña Rutas**

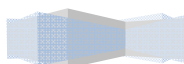
Desde esta pestaña el usuario podrá editar el mapa añadiéndole las rutas que desee. Si ya dispone de una ruta puede cargarla pulsando el botón *Cargar ruta*. Al pulsar este botón se le mostrará al usuario un cuadro de diálogo pidiéndole que indique qué archivo .rut desea cargar.



A continuación se detallan las funcionalidades del panel:

#### Dibujar ruta:

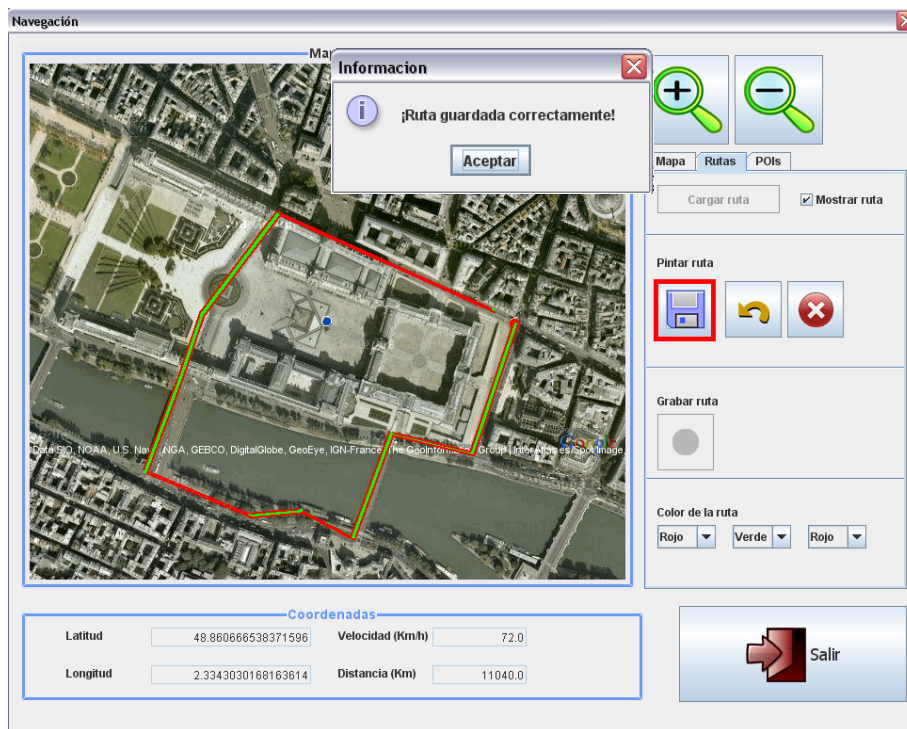
Esta funcionalidad está pensada para que el usuario cree a mano sus propias rutas dibujándolas con el puntero del ratón.

Pulse  para iniciar la ruta. El puntero del ratón adoptará la apariencia del icono. Haciendo click con el botón izquierdo sobre el mapa podrá ir generando la ruta que desee.


Pulse  para borrar el último tramo dibujado y  para cancelar completamente la ruta.



Por último pulse el botón  para guardar la ruta.

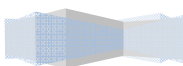


### Grabar Ruta:

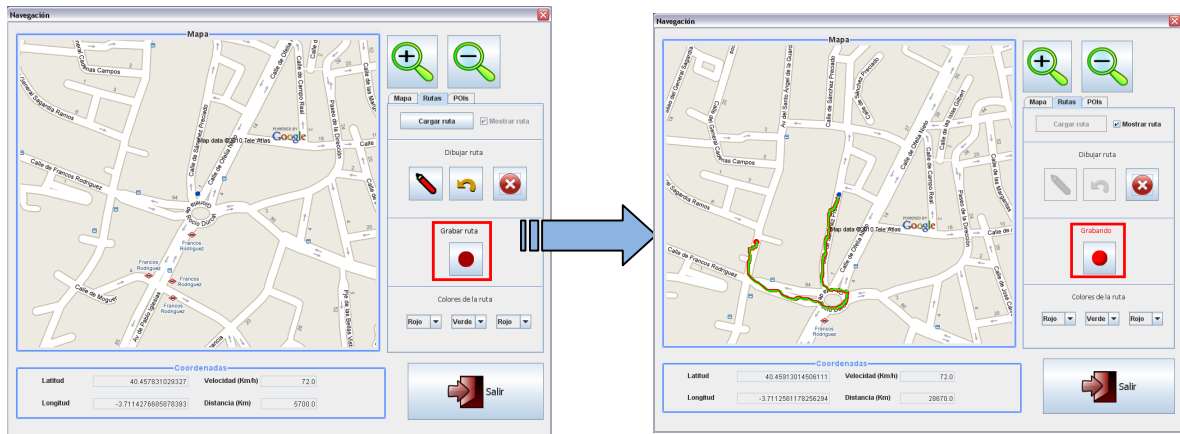
Pulsando este botón el usuario puede grabar sus propias rutas sin necesidad de pintarlas sobre el mapa tramo a tramo. Está pensada para usuarios que dispongan de dispositivo GPS, pero también ofrece la posibilidad de simular el movimiento pulsando el **botón derecho del ratón** sobre el mapa y desplazándolo sin soltarlo.

Tras pulsar el botón  el dispositivo GPS empezará a recibir datos sobre su posición. Recorra a pie o en algún medio de transporte la ruta que desee guardar, a medida que avance se irá pintando sobre el mapa el recorrido realizado.

Vuelva a pulsar  para guardar la ruta o  para cancelarla.







### Colores de la ruta:

Estos 3 menús desplegables permiten elegir el color de la franja izquierda, franja central y franja derecha respectivamente, de la ruta que se muestra por pantalla. Los colores a elegir son: *rojo*, *verde* y *azul*.

Esta funcionalidad es especialmente útil cuando el terreno tiene un color muy parecido al de ruta trazada, haciendo complicado distinguir entre ambos. Elija en cada caso los colores que más se adecuen a sus necesidades.

### Pestaña POIs

Esta pestaña permite al usuario editar su mapa añadiéndole puntos de interés, más conocidos como POIs (Points of interest). Los puntos de interés son lugares relevantes del mapa que el usuario desea recordar para posteriores navegaciones. Estos POIs pueden ser de distinto tipo dependiendo de su naturaleza. Más adelante se detallan los tipos que distingue Geomantes.

Además de crear sus propios puntos de interés, el usuario puede cargar otros que ya posea en un dispositivo Garmin o similar gracias al formato elegido para almacenar los POIs en Geomantes (.csv), pensando siempre la compatibilidad con el mayor número de plataformas. El formato .CSV es reconocido por numerosos dispositivos GPS haciendo portables los POIs guardados en Geomantes a otras plataformas y viceversa.

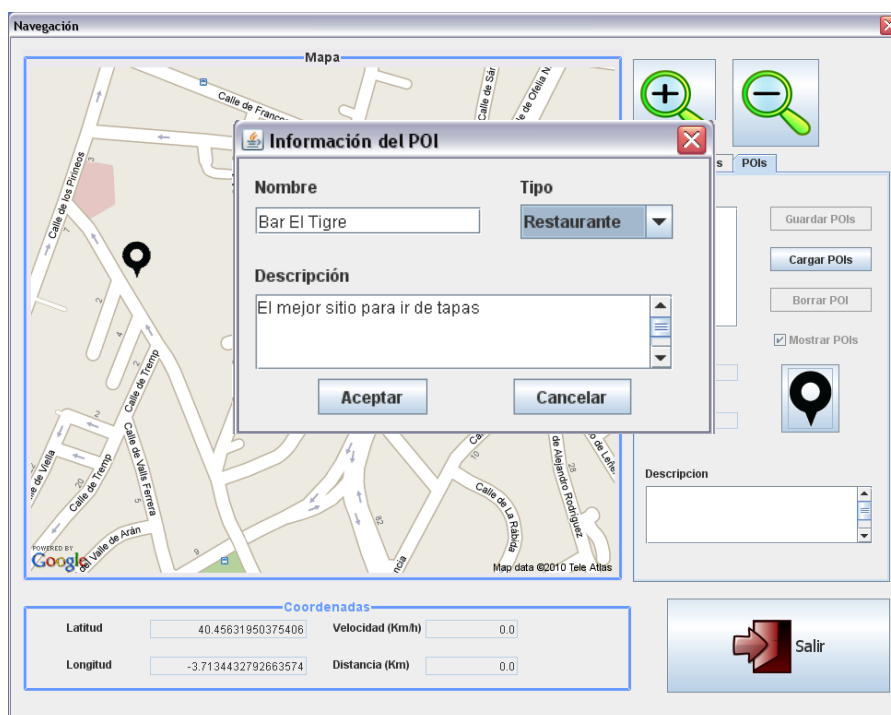


A continuación se detallan cada uno de los componentes de esta pestaña.

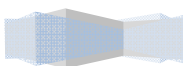
### **Botón**




Sirve para añadir un nuevo POI al mapa. Cuando se pulsa este botón el puntero del ratón adopta la forma del icono. Coloque la marca sobre el lugar del mapa deseado y pulse el botón izquierdo del ratón para añadir el nuevo punto de interés. Se le mostrará al usuario un cuadro de diálogo preguntándole por las características del POI como se muestra a continuación:



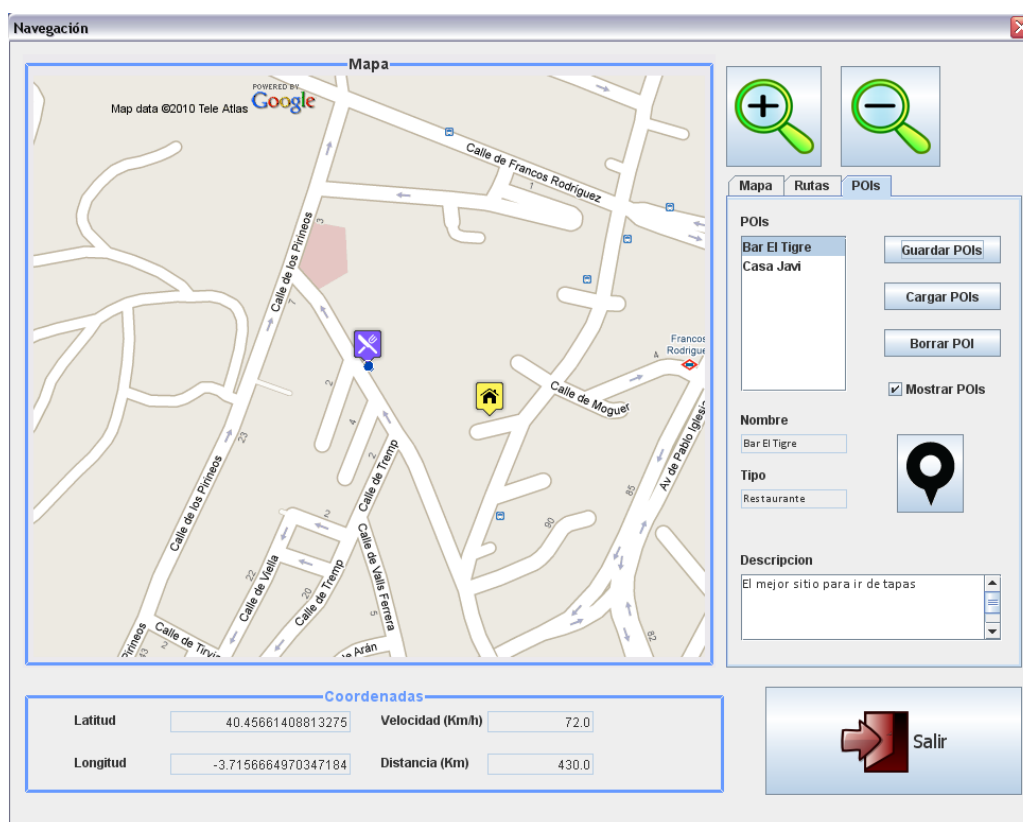
Escriba un nombre para el POI, una descripción y un tipo. Pulse Aceptar para añadir el POI o Cancelar para cerrar la ventana sin guardarlo.



A continuación se muestran los diferentes iconos mostrados para cada tipo:

Tipo	Dirección	Cafetería	Prims. Auxilios	Gasolinera	Hotel
Icono					
Tipo	Restaurante	Servicios	Parking	Otro	
Icono					

Tras añadir el POI aparecerá en el mapa el icono correspondiente a su categoría en el lugar indicado y su nombre en la lista de POIs.



La lista de POIs muestra el nombre de los puntos de interés añadidos hasta el momento. Al pulsar sobre un elemento de la lista se muestra la información



correspondiente al POI en los campos inferiores **Nombre**, **Tipo** y **Descripción** como muestra el ejemplo anterior. Puede borrar el POI seleccionado pulsando el botón de la derecha **Borrar POI** o dejar de visualizarlos momentáneamente en el mapa deseleccionando la casilla **Mostrar POIs**. Por último puede guardarlos para posteriores sesiones pulsando el botón **Guardar POIs**.

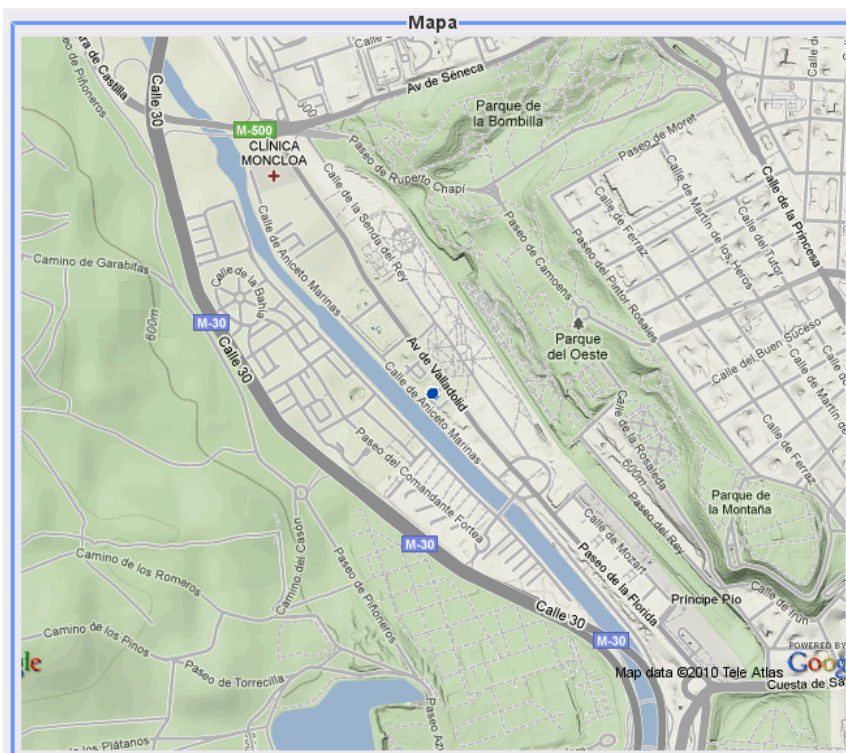
Este menú también ofrece la posibilidad de cargar POIs almacenados previamente en su PC pulsando el botón **Cargar POIs**. Se le mostrará al usuario un cuadro de dialogo pidiéndole que seleccione de entre sus archivos uno con extensión .csv. Este tipo de archivos puede obtenerlo habiendo guardado en sesiones anteriores una lista de POIs en Geomantes, puede descargarlo en internet desde numerosas páginas al ser un tipo de archivo muy utilizado o puede editarlo directamente a mano. La estructura de un archivo .csv ( Comma-Separated Values ) es la siguiente:

**\*\*Comentario\*\*Nombre del POI:Tipo:Descripcion,latitud,longitud**

Ejemplo:

**\*\*GEOPC\_POI\*\*Bar El Tigre:Restaurante:El mejor sitio para ir de tapas,40.45661408813275,-3.7156664970347184**

## Mapa

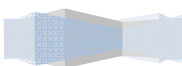
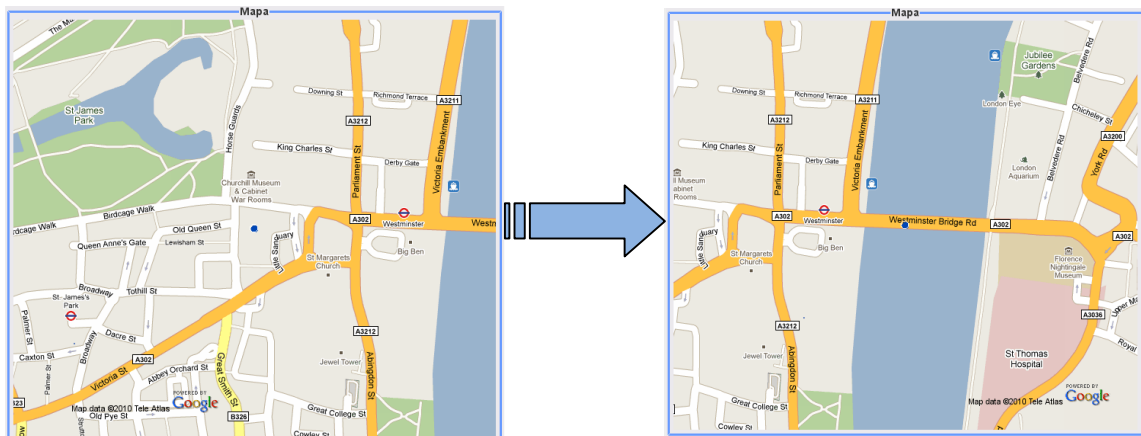


Elemento principal del menú. Consiste en un panel de tamaño 633 píxeles x 548 píxeles que va cambiando de apariencia respondiendo a los cambios solicitados por el usuario. Sobre los distintos tapices (mapa de carreteras, satélite, híbrido, terreno y mapas cargados por el usuario) que se van deslizando sobre el panel contiene un **punto azul** en el centro que simboliza al usuario. Además, el panel reconoce determinados eventos del ratón que se detallan a continuación:

### Pulsación continuada del botón derecho del ratón:

Coloque el puntero sobre el mapa y pulse el segundo botón. Sin soltarlo desplace el puntero en la dirección que desee para desplazar el mapa. Deje de presionar el botón para terminar el desplazamiento.

Esta funcionalidad es especialmente útil cuando desea visualizar zonas adyacentes al mapa actual ya que permite con un simple movimiento de ratón acceder a lugares que se encontraban fuera de los márgenes del mapa.



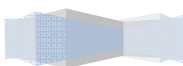
## Pulsación simple del botón derecho del ratón:

Coloque el puntero sobre la zona deseada del mapa y haga click con el segundo botón del ratón. Aparecerá un menú con dos opciones:

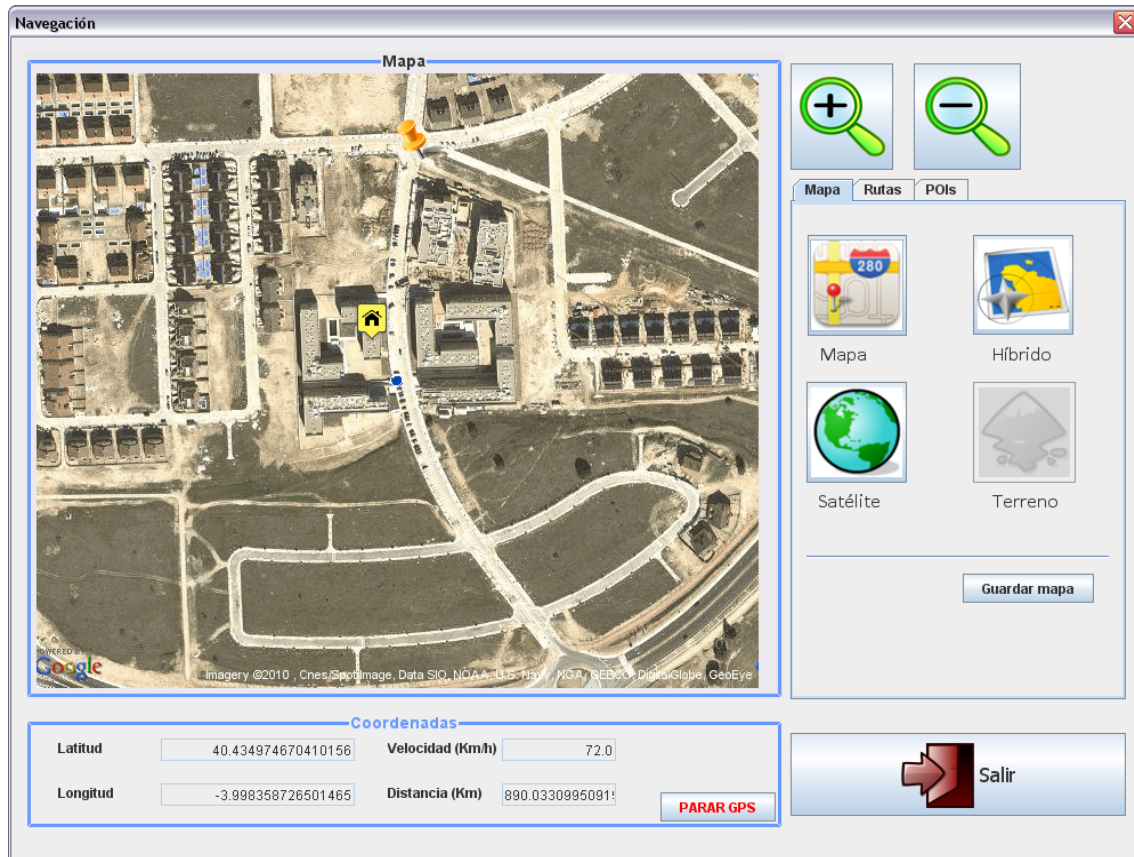


*Centrar Aquí:* coloca en el centro del mapa el lugar seleccionado con el ratón.

*Añadir POI:* forma alternativa de añadir un POI

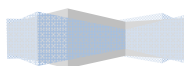


## Menú Navegación Libre



Este menú permite la navegación del usuario. Las posibilidades que ofrece son muy parecidas a las del menú anterior pero esta vez el mapa visualizado depende de la ubicación actual que proporciona el dispositivo GPS. El mapa mostrado se irá desplazando conforme al movimiento real del usuario para mantener a éste siempre en el centro de la imagen.

Al igual que en el menú editar mapas, el usuario puede hacer Zoom de la zona, visualizar distintos tipos de mapas y guardarlos, añadir rutas y POIs.



## Funcionamiento de aplicaciones externas

### ExtGps

La aplicación ExtGps se encarga de mandar los datos de posicionamiento del dispositivo móvil al ordenador a través de la tecnología Bluetooth. Una vez que se ha llevado a cabo la instalación en el móvil (ver sección Instalación), y se ha establecido la conexión vía Bluetooth entre el ordenador y el móvil, deberá esperar hasta que ExtGps obtenga señal Gps y en consecuencia pueda proporcionar datos de posicionamiento.

Para saber si todo funciona correctamente, deberá observar que, tanto la conexión vía Bluetooth como los datos del Gps, aparecen con un círculo verde a su izquierda.

### PoiConverter

GeomantesPC tiene dos formatos distintos para los puntos de interés:

- CSV (Comma separated value) con tres campos:

Longitud, Latitud, "Nombre"

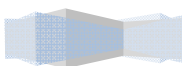
- POI (Punto de interés):

Longitud, Latitud, "Nombre", "Descripción", "Tipo"

Nuestra aplicación consigue interpretar ambos tipos de formatos, aunque a la hora de mostrar la información, en el caso de los puntos de interés con formato CSV lo único que aparecerá será el nombre. La idea es que GeomantesPC tenga su propio formato de puntos de interés (.poi) para que podamos añadirle la información que queramos en los campos descripción y tipo.

Debido a que hay algunos puntos de interés con formato CSV pero con campos extra, GeomantesPC los trata como si sólo tuvieran los tres campos mencionados anteriormente y en ese mismo orden. En caso de que el archivo con los puntos de interés no tenga el orden de campos anteriormente mencionado, no se podrán importar.

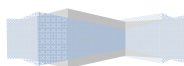
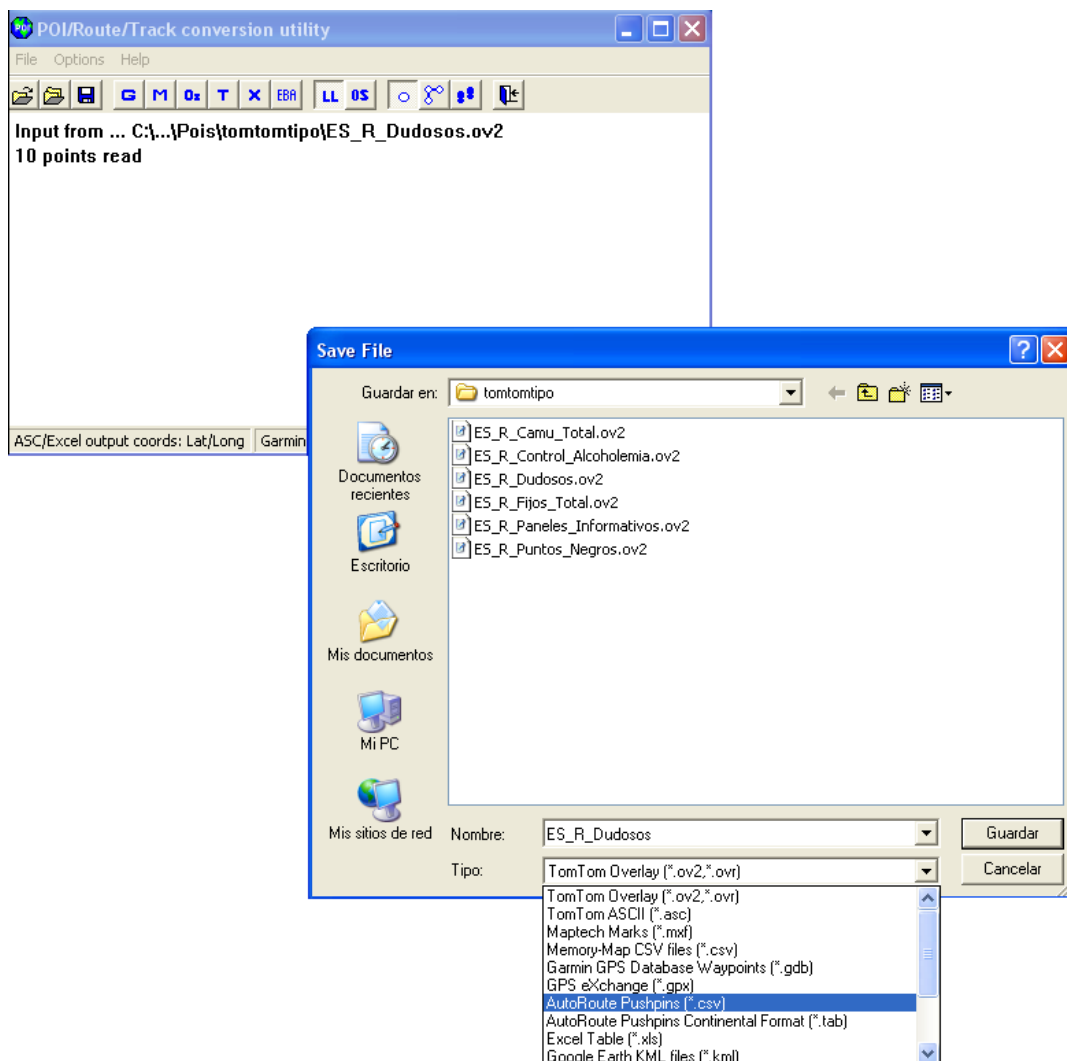
El programa POIConverter es capaz de convertir muchos formatos entre sí, como por ejemplo los de TomTom (.ov2), Garmin (.csv), etc. Para ello, únicamente tendremos que abrir el programa, y decirle tanto la fuente de los puntos de interés (y su formato) como el destino (y su formato).



El flujo es el siguiente:

- File/Open
- Seleccionamos el archivo fuente
- File/Save
- Seleccionamos el destino y el formato al que queremos convertir el archivo fuente.

- Como hemos comentado anteriormente, el formato .csv puede tener más de tres campos, incluso a veces tiene los campos en otro orden. Para que GeomantesPC pueda tratar correctamente los puntos de interés, el formato destino deberá ser AutoRoute Pushpins (\*.csv), tal y como se muestra en la imagen.





# CONCLUSIONES

## CONTINUIDAD DEL PROYECTO Y POSIBLES MEJORAS

Geomantes<sup>PC</sup> es la continuidad del proyecto *Geomantes*<sup>[1]</sup> desarrollado en 2008-2009, de ahí que queramos dedicar especial atención a este apartado e intentar facilitarle las cosas a un posible futuro equipo. Hemos sido muy cautelosos a la hora de documentar la aplicación y crear la memoria, además de haber establecido estructuras que posibilitan el seguimiento y registro de todos los avances. Valoramos en gran medida los esfuerzos realizados por el anterior grupo, pero desde el principio echamos en falta algo más de concreción precisamente en las posibles vías de desarrollo.

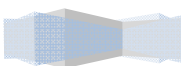
En consecuencia, queremos detallar un listado no sólo de características que puedan ser mejoradas o añadidas a nuestra aplicación, si no además una serie de alternativas de desarrollo que por su gran magnitud o falta de tiempo no hemos llevado a cabo.

1. **Mapas vectoriales.** Tal y como queda expuesto en la memoria de *Geomantes*<sup>[1]</sup>, esta característica ya fue apuntada por nuestros antecesores y vuelve a plantearse como una posible mejora. Las ventajas de trabajar sobre este tipo de mapas en lugar de utilizar *bitmaps*<sup>[2]</sup> son enormes y están muy relacionadas con la gestión de rutas. Son, entre otras, la posibilidad de crear rutas guiadas, calcular los caminos más cortos entre dos puntos y recalcular rutas sobre la marcha.

Su implantación fue estudiada por los miembros de este equipo al comienzo de curso y descartada por la falta de un proveedor de mapas vectoriales que permitiera descargar áreas de tamaño sustancial. Hasta donde hemos podido averiguar, [OpenStreetMap](#) es el proyecto más importante y de mayor envergadura ha nivel internacional, que además, se distribuye bajo licencia GNU. Desafortunadamente, por el momento no resulta viable su uso por lo anteriormente indicado, pero tiene el potencial suficiente como para que a medio plazo sea posible su uso y con ello, llevar a Geomantes<sup>PC</sup> a otra dimensión.

2. **BD de imágenes personales.** Aprovechando el mayor tamaño de pantalla, se trataría de crear las estructuras necesarias tanto para el almacenamiento como para la representación de imágenes asociadas a un punto geográfico. Desde un enfoque práctico, consideramos que esta mejora le podría resultar de especial interés a aquellos usuarios que deseen utilizar Geomantes<sup>PC</sup> como navegador *offroad*.

Ejemplo de uso sería tomar una instantánea y a través de Geomantes<sup>PC</sup>, geoposicionar la imagen y representarla en nuestro navegador. Lamentablemente, no hemos disfrutado del tiempo necesario para



incorporar esta característica a nuestra aplicación, pese a que con la edición de puntos de interés se pueden geolocalizar imágenes.

3. **Mejora en la calibración** añadiendo más de dos puntos a la hora de realizar la calibración.

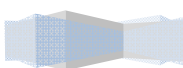
4. **Publicidad.** En la línea de lo anterior, sería posible introducir publicidad de establecimientos tales como restaurantes o gasolineras mediante la representación de imágenes con los logotipos e incluso eslóganes publicitarios de las firmas o compañías adscritas. Las posibilidades de crear conjuntos de empresas patrocinadas, lugares recomendados o favoritos, son algunas de las funciones que podrían aportar ayudas económicas con el fin de sufragar los gastos derivados del mantenimiento y desarrollo de Geomantes<sup>PC</sup>.

Además, existe la posibilidad de portar la aplicación a otras plataformas, utilizando los conceptos y todo el código reutilizable posible de *Geomantes* y Geomantes<sup>PC</sup>.

5. **Geomantes Android.** Apostamos por el despliegue de nuestro navegador GPS en una de las plataformas móviles que mayor desarrollo y crecimiento ha experimentado en los últimos tiempos. El número de ventas de estos terminales no ha parado de crecer desde su lanzamiento al mercado, y con él, se ha experimentado el nacimiento de una incipiente comunidad de desarrolladores.

Esta posibilidad fue contemplada por uno de los integrantes de nuestro grupo, pero la falta de experiencia unida al limitado tiempo de estudio de la materia, hizo inviable su desarrollo. No obstante, se tiene la firme creencia de que aunando esfuerzos desde el principio y empleando el tiempo necesario para familiarizarse con el entorno *Droid*, se pueden conseguir los conocimientos necesarios para alcanzar grandes logros y realizar una portabilidad completa y productiva.

Con estas líneas hemos querido esbozar un conjunto de vías maestras, pero ni son únicas ni exclusivas. Animamos a quienes tomen el testigo a explorar las posibilidades que este proyecto ofrece, indagar que productos existen en el mercado y considerar las mejoras propuestas o ampliar la familia de navegadores Geomantes.





## EXPERIENCIAS Y CONOCIMIENTOS ADQUIRIDOS

Valoramos muy positivamente el trabajo realizado, pero somos más conscientes que nunca de lo que supone afrontar una empresa de estas características. Como ya ha sido señalado anteriormente, hemos dado continuidad al trabajo realizado en 2008-2009 por otro grupo, lo que supone una inversión de tiempo considerable. Para estar listos y capacitados y modificar tan sólo una línea de código, no basta con una lectura superflua de la documentación disponible, sino por el contrario, hay que realizar una inmersión completa en un código ajeno y desconocido.

Cualquier programador con cierta experiencia sabrá ver que la labor de comprender el funcionamiento de una aplicación con miles de líneas de código, de cuyo desarrollo no haya sido participe, es una de las tareas más arduas y complejas que existen en programación. Ni que decir tiene que el hecho de tratarse de una aplicación para móviles en J2ME no ayuda precisamente a su comprensión.

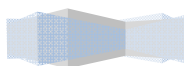
Pese a todo, *Geomantes*<sup>[1]</sup> ha resultado de gran ayuda en la creación de la versión para PC. El tiempo empleado en este estudio no ha sido en vano y aunque no hemos alcanzado un dominio tan alto como el de nuestros compañeros, hemos adquirido cierta experiencia en el desarrollo de aplicaciones móviles.

La práctica también nos ha enseñado que no es fácil decidir que vía de desarrollo se va a seguir en el proyecto y que hay que prestar especial atención a la hora de elegir, pues va a resultar determinante durante todo el proceso. De nuevo, el desperdicio de tiempo en barajar e investigar posibilidades que resultan a la postre descartadas, es cuantioso.

Por otro lado, la coordinación, la creación y gestión de las estructuras que dan soporte al desarrollo, la planificación o la asignación de tareas, son experiencias muy enriquecedoras a nivel personal y profesional. Hemos simulado, en nuestro último año de carrera, un entorno de trabajo semejante al que podemos encontrar en el mundo laboral, donde valores como el compromiso, la solidaridad o el compañerismo han estado presentes en todo momento. La planificación de reuniones ha sido otro de los puntos clave del éxito de *Geomantes*<sup>PC</sup>, ya que gracias a estas hemos podido llevar un control preciso y sistemático de los avances y trabajos llevados a cabo por cada uno de los integrantes.

Por último, pero no por ello menos importante, son los aspectos técnicos adquiridos, relacionados con la geolocalización, el uso de tecnologías que habilitan el geoposicionamiento y el mundo del GPS en general. Muchos de estos conceptos involucran áreas de conocimiento muy variadas y hasta ahora desconocidas para nosotros.

Por todo esto nos sentimos muy orgullosos de trabajo realizado y las metas alcanzadas, y esperamos tener la oportunidad de mostrar las habilidades y conocimientos que este proyecto nos ha brindado



# GLOSARIO

[1] Geomantes:

- Navegador GPS para telefonía móvil desarrollado para la asignatura de Sistemas Informáticos (Facultad de Informática de la Universidad Complutense de Madrid).

[2] Raster o bitmap:

- Rejilla rectangular de píxeles o mapa de bits.

[3] JVM:

- *Java Virtual Machine* o Máquina Virtual Java en español es una herramienta capaz de interpretar y ejecutar *bytecode* (o código binario generado por el compilador de lenguaje Java).

[4] Netbook:

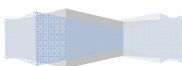
- Ordenador portátil de bajo coste, de reducido peso y dimensiones. El primero salió al mercado en 2007, cuando Asus presentó el ASUS Eee, de tan solo 0.9 Kg y 7" de longitud.

[5] NMEA 0183:

- Estándar que define por un lado la difusión de sentencias desde un Terminal a múltiples receptores y por otro la forma o contenido en que deben estar representados estos mensajes

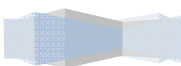
[6] PC:

- *Personal Computer*, habitualmente traducido como ordenador personal.

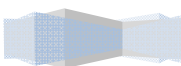


# BIBLIOGRAFÍA

- [1] Web 'Modelo vista controlador'  
[http://es.wikipedia.org/wiki/Modelo\\_Vista\\_Controlador](http://es.wikipedia.org/wiki/Modelo_Vista_Controlador)
- [2] Web 'Librería GPS Chaeron'  
<http://www.chaeron.com/gps.html>
- [3] Web 'Route 66'  
<http://www.66.com/route66>
- [4] Web 'Calidad del software'  
<http://www.calidaddelsoftware.com/>
- [5] Web 'Conferencia Calidad del Software. Universidad Nacional de la Pampa'  
[http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad\\_software.PDF](http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad_software.PDF)
- [6] Web 'Foro Sun'  
<http://forums.sun.com/index.jspa>
- [7] Web 'Proyecto fin de carrera: Gps y Java'  
<http://www.alu.ua.es/f/fjgr2/index.htm>
- [8] Web 'Api de Google Maps'  
<http://code.google.com/intl/es-ES/apis/maps/index.html>
- [9] Web 'Programas para GPS. Antonio R. Franco'  
<http://www.elgps.com/ftpgps.html>
- [10] Web ' ExtGps '  
<http://www.symarctic.com/beta/index.php?entry=entry070920-194854>
- [11] Web 'Formato ficheros Map'  
[http://www.rus-roads.ru/gps/help\\_ozimap\\_file\\_format.html](http://www.rus-roads.ru/gps/help_ozimap_file_format.html)
- [12] Web 'Programación: Patrón Singleton'  
<http://chuidiang.blogspot.com/2005/12/patrn-singleton.html>
- [13] Web 'OpenStreetMaps'  
<http://www.openstreetmap.org/>

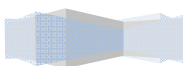


- [14] Web 'WGS84'  
<http://es.wikipedia.org/wiki/WGS84>
- [15] Web 'ElGps'  
<http://www.elgps.com/ftpgps.html>
- [16] Web 'Cartesia: Calibración de mapas'  
<http://www.cartesia.org/foro/viewtopic.php?p=24124>
- [17] Web 'Instalación javax.comm'  
<http://lefun.es.wordpress.com/2009/02/27/instalacion-del-java-comm-api-en-windows/>
- [18] Web 'Poi Converter'  
<http://rjdavies.users.btopenworld.com/html/poiconverter.html>
- [19] Web 'Cómo calibrar un mapa con OziExporer'  
<http://www.elgps.com/ozi/calibrarmapa/calibrarconozi.html>
- [20] Libro Antonio, MORENO JIMÉNEZ. *Sistemas y Análisis de la Información Geográfica , Manual de autoaprendizaje con ArcGIS*. 1ª ed. Paracuellos de Jarama, Madrid: RA-MA, 2006, 895 p.
- [21] Libro Monserrat GÓMEZ DELGADO y Jose Ingacio BARREDO CANO. *Sistemas de Información Geográfica y evaluación multicriterio en la ordenación del territorio*. 2ª ed. Paracuellos de Jarama, Madrid: RA-MA, 2005, 279 p.



## PALABRAS CLAVE

- GeomantesPC
- GPS
- Bluetooth
- Raster
- JavaSE
- WGS84
- GoogleMaps



*Los alumnos que hemos desarrollado este proyecto autorizamos a la Universidad Complutense de Madrid a difundir y a utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la propia memoria como el código, la documentación y/o el prototipo desarrollado.*

Alfonso Jiménez Regalado

Javier Sangrador Martín

Alejandro Barrera de Madariaga

